

Dudley Knox Library, NPS
Monterey, CA 93943

T211536

**Approved for public release;
distribution unlimited.**

FORTRAN INTERACTIVE
ROAD DESIGN

A Special Research Problem
Presented To
The Faculty of the School of Civil Engineering
Georgia Institute of Technology

By

Joseph P. Cuccu

In Partial Fulfillment
Of the Requirements for the Degree of
Master of Science in Civil Engineering

ABSTRACT

This report provides a prototype computer program for the design of roads in an Amphibious Operations Area for the Marine Amphibious Force. The report is divided into two sections; the State of the Art Research, and the Road Design Program Development.

The State of the Art section of this research provides a background of military construction management programs developed for microprocessors. These are reviewed in the first five chapters: OVERVIEW, ARMY PROGRAMS, NAVY PROGRAMS, MARINE CORPS PROGRAMS, and GEORGIA TECH PROGRAMS. The State of the Art in relation to this research problem is the Road Design Program of LCDR R. R. Setzekorn.

The development of a Road Design computer program is the subject of the final two chapters: PROGRAM DESIGN and CONCLUSIONS. Here the design of a road is broken down into eight functional stages; Horizontal Alignment, Superelevation, Soils, Vertical Alignment, Cross Section, Intersection Points, Cross Sectional Area, and Mass Balance. These stages are, for the most part, based on procedures developed in Army Technical Manuals. However, some design methods from Surveying, by Harry Bouchard, et. al., and Highway Engineering 4e, by Paul H. Wright and Radnor J. Paquette; are also used. The flow charts and the final prototype computer program, Fortran Interactive Road Design (FIRD), were developed by the author of this report. Input to this program is interactive with the user, and requires the use of the user's manual. Output is received after completion of each of the eight functional design stages.

ACKNOWLEDGEMENT

I wish to express my appreciation to my wife, Sherrie, for her contributions in the development of the flow charts. I also wish to express my gratitude to Ms. Marge McMullen, whose consummate secretarial skills produced this report.

TABLE OF CONTENTS

I. OVERVIEW 1

II. ARMY CONSTRUCTION MANAGEMENT PROGRAMS 4
U.S. Army Engineering Center
20th Engineer Brigade
35th Signal Brigade

III. NAVY CONSTRUCTION MANAGEMENT PROGRAMS 9
31st Naval Construction Regiment

IV. MARINE CORPS PROGRAMS 19
Amphibious Logistic Support Ashore System

V. GEORGIA INSTITUTE OF TECHNOLOGY CM PROGRAMS 34
Military Water Points
Computerized Network Analysis
Computer Aided Designs for MAF Engineers
Design of Heliports and Expeditionary Airfields
MAF Equipment Scheduling System
Road Design Procedures for Microprocessors

VI. PROGRAM DESIGN 51
Horizontal Alignment
Superelevation
Soils
Vertical Alignment
Cross Section
Intersection Points
Cross Sectional Area
Mass Balance

VII. CONCLUSIONS 79
Synopsis
Program Limitations
Future Enhancements

REFERENCES 88

APPENDICIES

A. Programs Developed by U.S. Army
Engineer School 86

B. Programs Developed by 20th Engineer
Brigade 88

C. User's Manual for Program FIRD 90

D. FIRD Program Flow Chart 123

E. FIRD Program Listing 144

F. Sample Run of Program FIRD 186

CHAPTER I

OVERVIEW

In any military conflict the capability of providing adequate construction support is a virtual necessity. Most of the design procedures for this construction support are well delineated by numerous Army, Navy, and Marine Field and Technical manuals. Unfortunately, in an actual battlefield situation, the military engineer will have little time to research these manuals to develop the required design. Foreseeing this, computer aided designs (CAD) have been developed for the use of the combat engineer.

Initial work on the development of computer aided designs for combat engineers began at the U.S. Army Engineer Center, Fort Belvoir, Virginia, in the 1960's. Their research evolved around the use of small rugged microcomputers that could be used by mobile engineering units. Though this research was later discontinued, recent improvements in microcomputers have renewed interest in computer aided designs that could be used at the Battalion Field Level.

In the 1970's the 20th Engineer Brigade of the XVII Airborne Corps at Ft. Bragg, North Carolina began development of programs for their microcomputer. The 20th Engineer Brigade tailored the programs developed by the Fort Belvoir School to meet their needs. In addition, the 20th Engineer Brigade developed their own software packages to aid their own personnel, security, and logistics

management needs, as well as their construction operations mission. Unfortunately the sensitivity of the 20th Engineer Brigade's microcomputer system is such that the current system cannot withstand field conditions.

The U.S. Army Signal Corps also began development of a microcomputer system in the early 1970's. The Electromagnetic Compatibility Analysis Center of the 35th Signal Brigade developed software to plan and manage communications-electronics (C-E) engineering systems in support of major field exercises. As with the 20th Engineer Brigade, the 35th Signal Brigade does not plan to deploy the microcomputer system due to the system's sensitivity.

The Naval Construction Force also procured a microcomputer system during the 1970's. Software programs, developed by the Civil Engineer Support Office (CESO), Port Hueneme, California was promulgated to each of the permanent deployment sites of the eight Naval Mobile Construction Battalions (NMCB). The computer programs developed to date include a CPM program, two personnel management programs, and programs for project material and tool inventory control. This microcomputer system, like the others, is for peacetime application only and is not part of the NMCB's table of allowance.

In 1974 the Marine Corps procured a hardened version of the IBM Series 1 microcomputer to automate processing of their administrative data. A software package, the Automated Data Processing Equipment-Fleet Marine Force (ADPE-FMF) program, was developed to efficiently transfer personnel, financial,

operational and logistical data from the battalion or divisional level to a centralized main frame computer. The ADPE-FMF program corrects data entries and stores them directly on a disk. This disk is then physically carried to the main frame computer site and read into the larger system's data bank. The ADPE-FMF system, though not oriented toward combat engineering, is the only active system currently considered deployable by the Armed Services.

Finally prototype programs, designed for microcomputer implementation by military engineers, have been developed by the Civil Engineering Laboratory, Port Hueneme, California and the Georgia Institute of Technology. One of these programs, "Computer Aided Road Design Procedures for Microprocessors" by Robert R. Setzekorn (see reference (1)) is an interactive program for the design of an improved road. It was written in Cyber 70 basic language. The design program developed herein is based on the research of LCDR Setzekorn. The intent of this research is to develop a Fortran IV prototype road design program which meets and extends the capabilities of the Setzekorn program.

CHAPTER II

ARMY CONSTRUCTION MANAGEMENT PROGRAMS

The first to recognize the construction problems peculiar to combat engineers was the U.S. Army Engineering Center, Fort Belvoir, Virginia. During the 1960's the school was provided a WANG 2200 VP microcomputer. Using this system, the school developed prototype programs in the Basic language to aid their combat engineering units. A listing of these programs and their capabilities is given by Appendix A.² This system was later abandoned by the school in as much as the computer system utilized was not rugged enough to operate under field conditions. Nevertheless recent improvements in microcomputers have renewed interest in computer aided designs that could be used at the Battalion Field level. In addition, the school is studying the possibility of using programmable hand held calculators with preprogrammed magnetic cards to be used at the company level or below. This system however, even if applied, will not be able to handle the vast majority of the construction design tasks required by Army Combat Engineering Units.

The 20th Engineer Brigade of the XVIII Airborne Corps at Ft. Bragg, North Carolina was also provided a WANG 2200 VP. In addition, the Brigade was given the Ft. Belvoir school's prototype programs, which the Brigade later tailored in order to meet their needs. Using the WANG system, the 20th Engineer

Brigade began development of their own computer aided designs. A listing of the 20th Engineer Brigade CADs is given by Appendix B. As noted on Appendix B, many of these programs do not deal with the Brigade's primary mission of combat engineering, but instead were developed to aid the unit's personnel, security, and logistic management effort. The 20th Engineer Brigade currently quarters its WANG computer system in a portable trailer. Though this trailer is a controlled environment, the WANG's sensitivity to damage by fine dust is such that the system cannot be deployed. Presently the 20th Engineer Brigade is attempting to replace the WANG system with a more rugged system.

The 35th Signal Brigade also began CAD development on a WANG 2200 VP during the 1970's. As the 35th Signal Brigade's primary mission deals with communications, their program development is not pertinent to this research. Nevertheless, it is of interest to note that the 35th Signal Brigade also feels the WANG 2200 VP computer system is too sensitive to operate under field conditions. Both the 20th Engineer Brigade and the 35th Signal Brigade are attempting to procure a new system rugged enough to be deployable. The system desired by both these units is the ONYX Z8002 microcomputer system.²⁺³ This system is shown by Table 2.1. Nevertheless, as the systems will be procured independently there is no guarantee that either unit will actually get the desired system.

Another interesting program of the Army Signal Corps is the Army Tactical Frequency Engineering System (ATFES).⁴ It's

TABLE 2.1

Computer System Components for the 20th Engineer
Brigade and the 35th Signal Brigade

- * ONYX Z8002 Micro computer with 128KB memory
and 10 serial ports
- * ONYX Magnetic tape controller
- * 9-track tape drive formatter, 800/1600 BPI, 45 IPS
- * Data Media VT-82, interactive terminals
- * High speed printer
- * Daisey printer/plotter

function will be to plan and manage communications in a manner similar to the 35th Signal Brigade's existing software package. The microcomputer system chosen for this program is given by Table 2.2.⁴ The plan is to deploy this microcomputer system to four permanent deployment sites in Europe. This system promises to be even more effective in the future due to a tri-service commitment to standardize software and hardware for computer processing of telecommunications. This program, known as Joint Tactical Communications (TRI-TAC),⁵ may even be extended to include our allies in NATO. The standardization of computer systems is an important step toward efficient use of our computer data bases.

TABLE 2.2

ATFES Pilot Program Hardware

- * PDP 11/70 Processor 2/512 KB memory (RSX11M + operating system)
- * RWM03 67MB disk drive
- * TWE 16 9-track tape drive 800/1600 BPI, 45 IPS
- * VT 100 Alphanumeric display and keyboard
- * LP11-VA Line printer, 300 LPM
- * LXY 11-AD Printer w/OCR capability
- * LA 120 Character printer, 180 CPS
- * TEKTRONIX 4027 color graphic display and keyboard
- * VERSATEC 1200A Printer/plotter, 1000 LPM/200 DPI
- * MODEM
- * Encryption devices
- * Teletypewriters

CHAPTER III

NAVY CONSTRUCTION MANAGEMENT PROGRAMS

In the 1970's the 31st Naval Construction Regiment (31st NCR) of the Naval Construction Force (NCF) began implementation of an automated construction management system to aid the construction planning of deployed construction battalions. By this system, each deployed battalion would submit arrow diagram CPM's of their primary construction projects to the 31st NCR for breakdown and input into a main frame computer at the Naval Construction Battalion Center,⁶ Port Hueneme, California. A summarized output would then be shipped back to each construction battalion for review. Updating was handled in the same manner.

As this system proved untimely and inefficient, the Civil Engineering Support Office (CESO), Port Hueneme was tasked with development of a deployable microcomputer system to be used by the construction battalions. To this purpose the microcomputer system of Table 3.1 was purchased, and CESO began development of prototype programs.⁷ Currently the following programs are available to deployed construction battalions:

1. Construction Management (CM-4) program
2. Material Liason Office program
3. People program
4. Medical program

TABLE 3.1

Naval Construction Force PDP 11/03

Minicomputer System Components

- * LSI-11/2 CPU
- * RT-11 64 K byte floppy disk-based system
(with dual RX02's)
- * Four channel Asynchronous serial line unit
- * Bootstrap loader/diagnostic/terminator/
ROM module
- * Nine slot box
- * Hard disk subsystem with add on disk drive
- * LA 38 DECwriter IV terminal
- * VT 100 video terminal

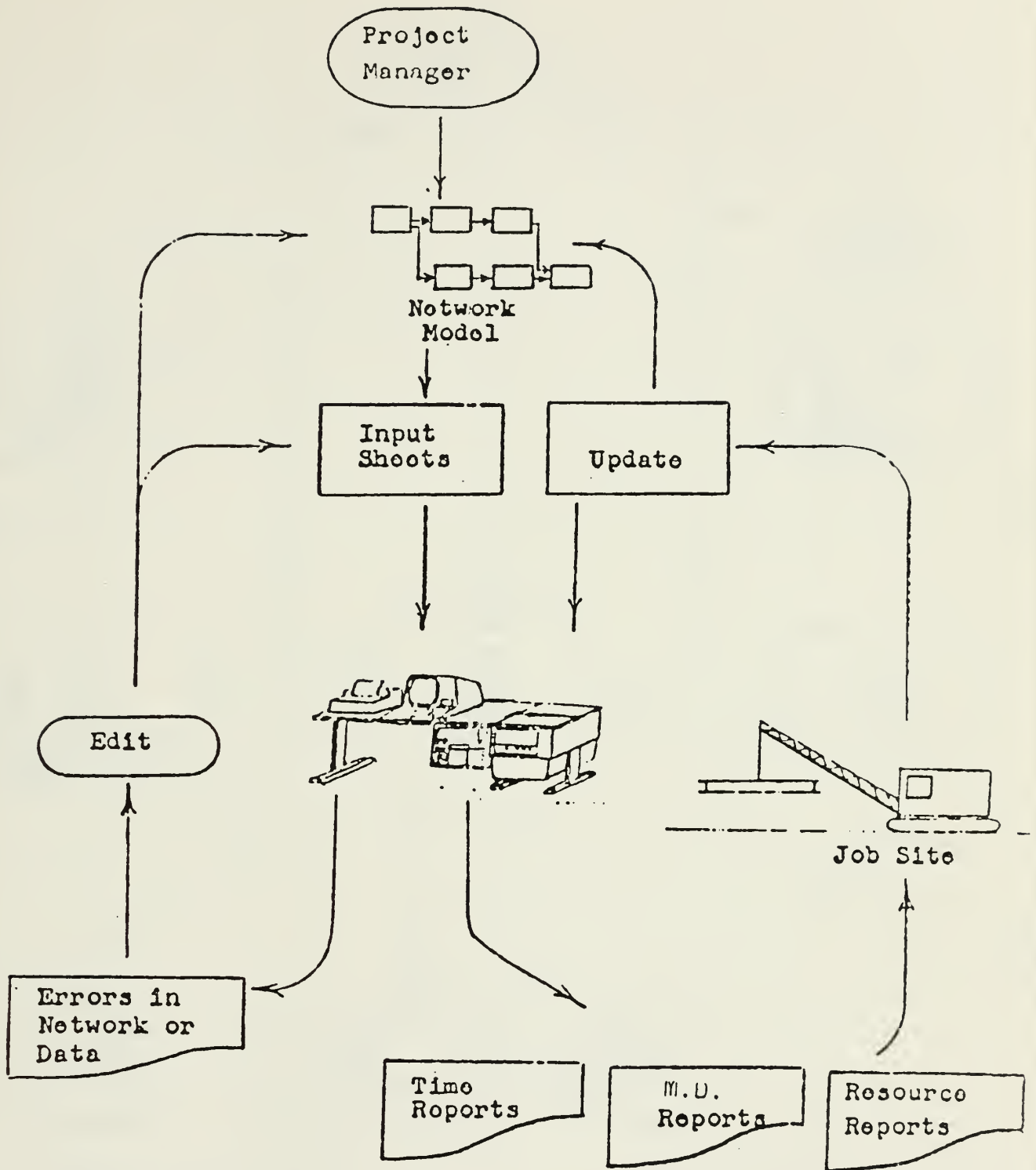
5. Tool program

Each of these programs were developed with interactive formating. They were designed so that even Seabees, without a programming background, could do the inputing. Nevertheless, both Construction Regiments have programs to train battalion personnel during each battalion's homeport (training) period.

The user of these programs responds to the computer's input data questions on a video display. The output reports are generated on a line printer. The output is then shipped to the Construction Regiments in monthly progress reports.

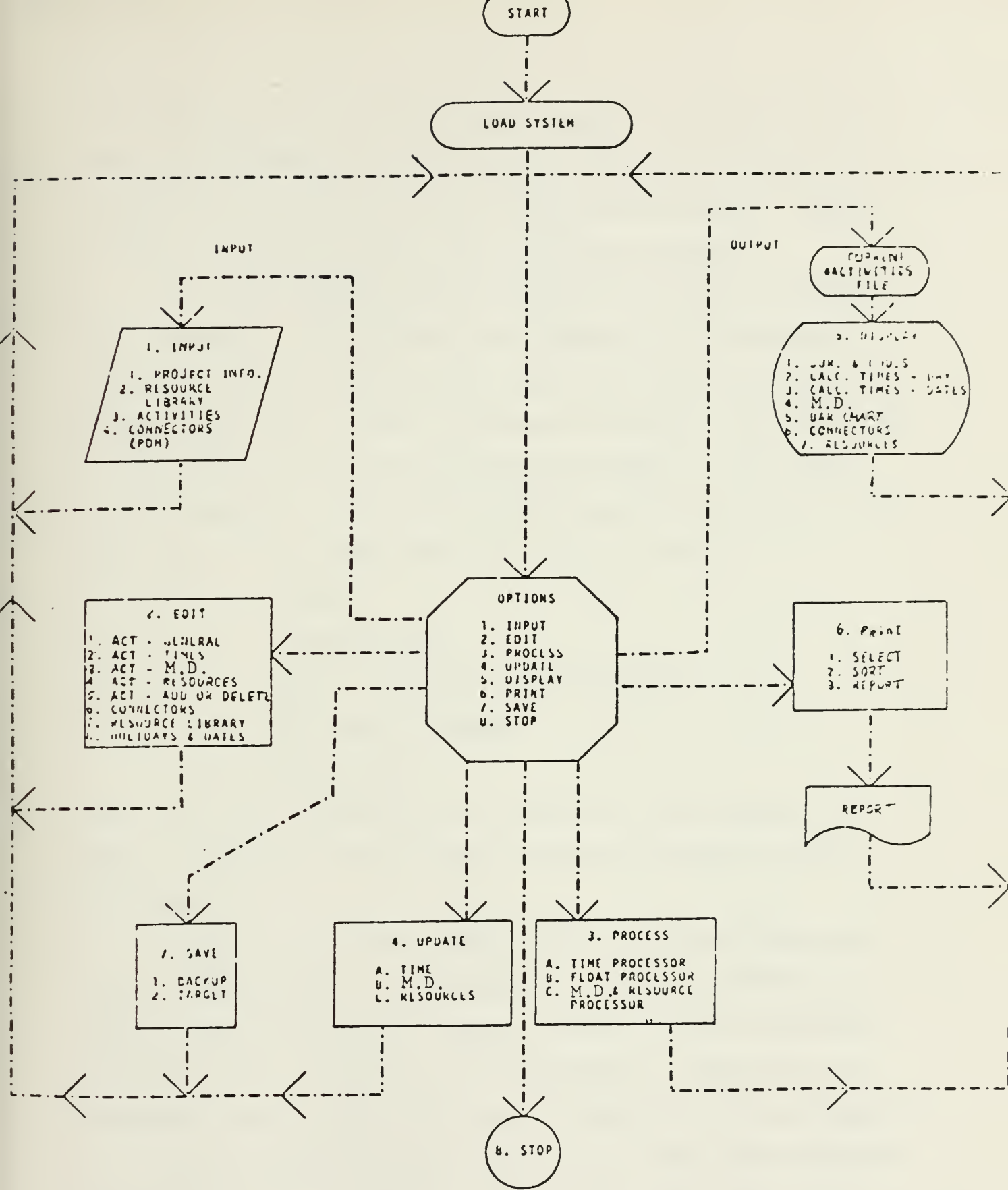
The Construction Management program was developed by Structural Programming, Inc. It is a basic network scheduling program capable of handling both arrow and precedence notation. The user can define a calendar and impose milestone date schedules on the network. Up to 99 resources can be assigned to any one network with up to six resources assigned to a specific activity. As resource balance, rather than cost control, is of the utmost concern to a battalion, the resource assignment allows for display of resource requirements. After input of the data, editing procedures are available to allow review and correction of the input data. A variety of line printer outputs is available from this program. The flow charts on this software package are given by Figures 3.2 and 3.3.⁸

The Material Liason Office (MLO) program allows the battalion to moniter and manage construction materials. The program inventories materials on site and on order, gives status of materials on order, and cross references these materials with



CM-4 System Flow Chart

Figure 3.2



CM-4 Program Flow Chart
Figure 3.3

the various projects. The program also has the capability of recording the demand of more common construction material requests so that proper inventory levels are maintained. Figures 3.4 and 3.5 give the flow charts for this program.⁹

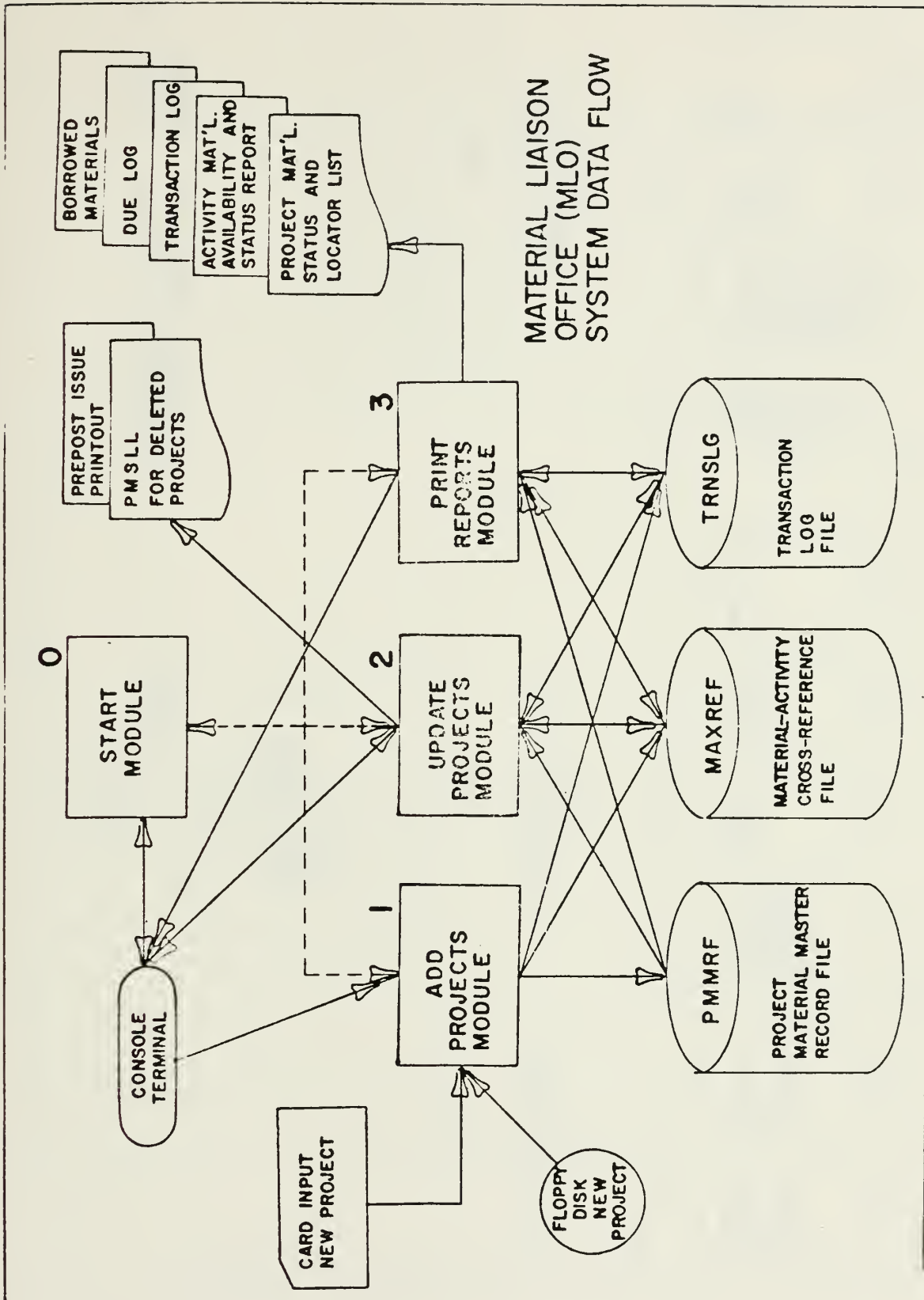
The People program was developed for management of battalion personnel. The program input consists of information on individuals such as name, social security number, address, pay-grade, rate, date of birth, training and skills. Up to 24 items of data can be associated with each individual. The program output reports on this stored information sorted in up to nine data items. The program has five built in reports:

1. Roster
2. Address roster
3. Roster with PRCP (training) skills
4. PRCP skills summary
5. Short roster

This data can be updated in the edit mode of the program. A menu of functions that the People program performs is given by Figure 3.6.¹⁰

The Medical program manages medical records of personnel within the battalion. The program stores such information as blood type, inoculations, medical examination and urinalysis testing on each individual in the battalion. There are four different output reports that can be generated by this program.¹¹

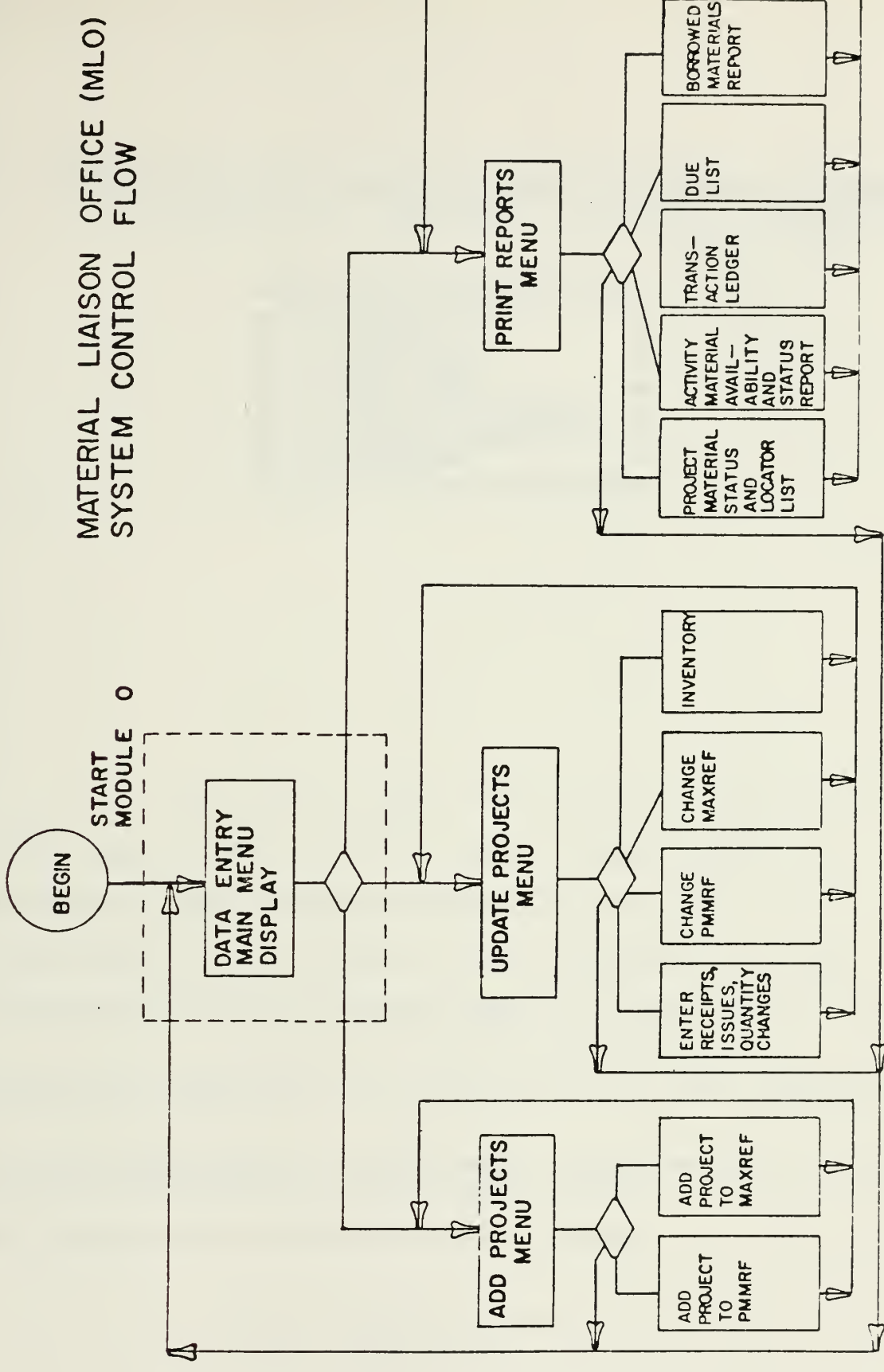
The Tool program manages the tool and small power equipment allowances for the battalion. Up to seven data items can be assigned to any specific tool. Data items include such infor-



MATERIAL LIAISON
OFFICE (MLO)
SYSTEM DATA FLOW

MLO System Flow Chart
Figure 3.4

MATERIAL LIAISON OFFICE (MLO)
SYSTEM CONTROL FLOW



MLO Program Flow Chart
Figure 3.5

KNOW PEOPLE PROGRAM
MAIN MENU

```
*****  
** 1. DELETE PEOPLE **  
** 2. ADD PEOPLE **  
** 3. UPDATE PEOPLE **  
** 4. PRINT REPORTS **  
** 5. SERVICE PRCP SKILLS FILE **  
** 6. LOAD PEOPLE DATA BASE **  
** 0. END PROGRAM **  
*****
```

Delete People provides the means to delete a record from the file.

Add People provides the means to add records to the file.

Update People provides the means to change information in a record.

Print Reports generates a hard copy report fo all data that is in the file.
This report can be generated in sort sequences wanted.

Service PRCP Skills File provides the means to print, add, delete, or renumber
the PRCP skills file.

Load People Data Base will transfer packed and unpacked data files (from GEMS)
to unpacked format on hard disk or floppy diskette.

End Program provides the means to halt the program.

People Program Main Menu
Figure 3.6

mation as tool custody, last preventive maintenance and safety inspection, and condition. The output report lists the tools and data, flagging those tools overdue for custody or inspection.¹²

The Naval Construction Force is currently developing more programs to aid their construction effort. Nevertheless, due to the sensitivity of the microcomputer system, only peacetime applications of the system are being considered.

CHAPTER IV

MARINE CORPS PROGRAMS

Although not construction oriented, the Marine Corps has a hardened version of the IBM Series 1 microcomputer which they have used to automate processing of their administrative data. The components that make up of the IBM microcomputer system is shown by Figure 4.1.¹³ The Automated Data Processing Equipment-Fleet Marine Force (ADPE-FMF)¹³ software package is used to efficiently transfer personnel, financial, operational and logistical data from each battalion's IBM microcomputer to a centralized main frame computer held at division headquarters. Battalion input data is entered into the microcomputer which stores this information on a disk. This disk is then physically shipped to the main frame computer site and read into the larger system's data bank.

Though this system sounds inefficient, a system quite similar to the Marine ADPE-FMF is currently under development by the Army. The Division Level Data Entry Device (DLDED)¹⁴ program would put microcomputers on a company level so that administrative and logistical requirements of that unit could be inputted. This data would be stored by each microcomputer on a disk and either shipped to the main-frame computer or transmitted by the aforementioned TRI-TAC system. The Army feels that such a system could eventually support many different types of forward units, perhaps even engineering units. On the other hand, the

TABLE 4.1

ADPE-FMF Equipment Component Listing

1. IBM 4952 Central Processor Unit with 64K bytes of memory
2. IBM 4978 Display Station with 9 inch Video Screen and Keyboard
3. IBM 4974 Terminal prints 120 characters per second
- *4. IBM 4469 Magnetic Tape Drive Unit
- **5. IBM 4470 Paper Tape Punch

* Provided only to units requiring this type of interface.

** Provided to units embarked on Navy vessels to interface Navy communication equipment.

Marine Corps will concentrate its development of the ADPE-FMF system on administrative and operational needs.

The Marine Corps, however, is not insensitive to the needs of their combat engineering units, and is fully aware of the magnitude of both the construction and logistical support it will require during an amphibious operation.¹⁵ To this purpose the Amphibious Logistic Support Ashore (ALSA) system was developed by the Marine Corps.

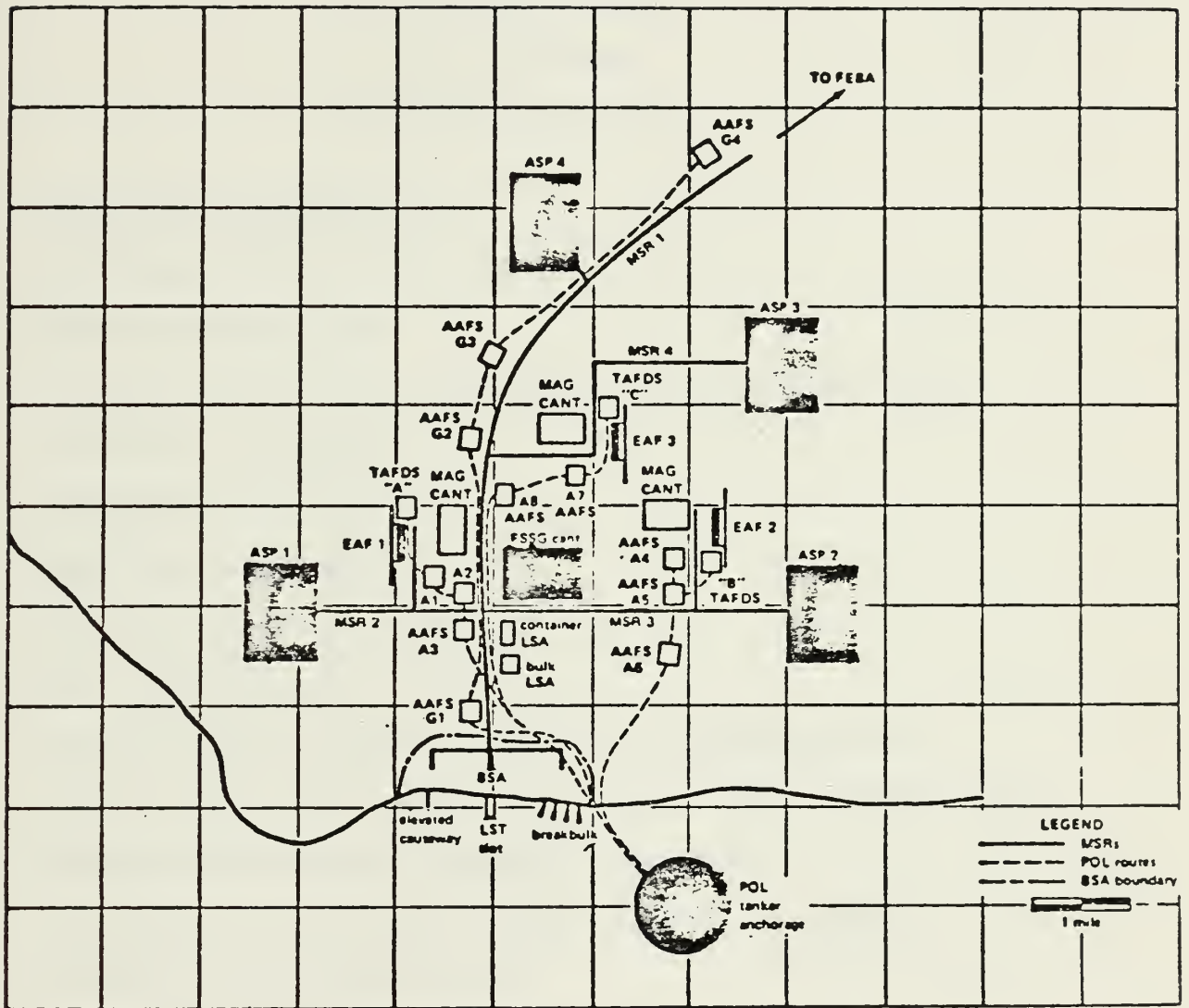
The ALSA system is structured to provide a continuous stream of supplies, equipment and construction support to a Marine Amphibious Force (MAF) unit engaged in securing a hostile beach-head. The engineering units of a Marine Air/Around Task Force (MAATF) unit will utilize the ALSA system to provide the construction support. The primary construction requirement of the Marine Air/Around Task Force is the construction of the Amphibious Objective Area (AOA) or advanced base. This is no small feat. Minimum requirements of an AOA are as follows:

1. Dependable main supply routes linking beach support areas (BSAs), logistic supply areas, ammunition storage points (ASPs), and combat troops.
2. Logistic supply areas for the storage, breakout and retrograding of containers and the storage of breakbulk supply items.
3. Ammunition supply points with protective earth revetments for the storage of sufficient quantities of ordinance to supply three Marine Air Groups and one Marine Division.

4. POL storage (AAFS), distributing and issuing systems (TAFDS, SIXCON).
5. Berthing, messing, administrative and work areas, and facilities to support 8,800 combat service support personnel assigned to the MAF Force Troops and 15,000 personnel assigned to the Marine Air Wing (MAW).
6. Expedient airfields to support the MAW (without benefit of existing facilities, at least three expeditionary airfields and several V/STOL and VTOL pads would have to be constructed to move the MAW ashore. A multitude of support facilities associated with these airfields must be available for the airfields to achieve a sustained operational capability.)"¹⁶

Figure 4.2¹⁶ is a representation of a typical Amphibious Operations Area layout. Note that the AOA land requirements is almost 56 square miles. This tends to highlight the magnitude of the horizontal construction effort required of the MAGTF engineer.

The Marine Corps recognized that many of the separate construction tasks of its engineering units could be systematically designed and quantified. To this end the Civil Engineering Laboratory (CEL), Port Hueneme, California was tasked with a systems study of the MAGTF engineer's taskings.¹⁷ Through this, CEL concluded that many of the separate construction tasks could be planned through the use of a small, portable computer system. Tables 4.3 and 4.4¹⁷ define those tasks that might be planned with computer aided designs. CEL is currently devel-



Typical AOA Layout

Figure 4.2

TABLE 4.3

U.S. MARINE CORPS
M.A.F. ENGINEER
MAJOR TYPES OF TASKS

1. ENGINEER RECONNAISSANCE
2. OBSTACLES AND/OR MINES BREACHING
3. DEMOLITION MISSIONS
- * 4. OBSTACLE FABRICATION AND INSTALLATION OR POSITIONING
5. MINEFIELD CLEARANCE
- * 6. MINEFIELD PLACEMENT
7. BOBYTRAP PLACEMENT
8. ROAD REPAIR AND MAINTENANCE
- * 9. ROAD CONSTRUCTION
10. BRIDGE REINFORCEMENT - (OTHER THAN PREFABRICATED)
11. BRIDGE REPAIR AND MAINTENANCE (OTHER THAN PREFABRICATED)
- *12. PREFABRICATED BRIDGE ERECTION (STANDARD)
- *13. PREFABRICATED FLOATING BRIDGE ERECTION (STANDARD)
- *14. TIMBER BRIDGE CONSTRUCTION
15. PREFABRICATED BRIDGE MAINTENANCE
16. BRIDGE REPLACEMENT (PREFAB. WITH SEMI-PERMANENT BRIDGING)
- *17. RAFT CONSTRUCTION AND OPERATION
- *18. EXPEDITIONARY AIRFIELD CONSTRUCTION
19. EXPEDITIONARY AIRFIELD MAINTENANCE
- *20. HELICOPTER LANDING SITE AND ZONE PREPARATION

*These tasks are those which can be planned by computer.

TABLE 4.4

MAJOR TYPES OF TASKS

- *21. POTABLE WATER SUPPLY INSTALLATION
- *22. HYGIENIC SERVICES INSTALLATION
- *23. ELECTRICAL SERVICES INSTALLATION
- 24. CUT AND COVER TYPE TEMPORARY FORTIFICATIONS CONSTRUCTION
- *25. CAMP CONSTRUCTION
- 26. WAREHOUSE REPAIR AND MAINTENANCE
- 27. DRAINAGE SYSTEM CONSTRUCTION
- 28. DRAINAGE SYSTEM MAINTENANCE AND IMPROVEMENT
- 29. CAMOUFLAGE INSTALLATION
- *30. FUEL STORAGE & DISTRIBUTION SYSTEM INSTALLATION
- 31. FUEL STORAGE & DISTRIBUTION SYSTEM INSTALLATION MAINTENANCE AND OPERATION

*These tasks are those which can be planned by computer.

oping a prototype software package of programs as part of its Development Engineering Planning (DEP) system. As part of this development process, CEL contracted with the Georgia Institute of Technology to study the existing level of development of MAF suitable design support systems and to propose methods for implementing design support systems in the MAF mission situation. The prototype microcomputer programs that evolved from this contract will be discussed in the next section.

The microcomputer system CEL is developing the DEP system on is the Tektronix 4052 Graphic System. CEL had previously used this hardware in the development of the Amphibious Objective Area Land Management System (AOALMS).

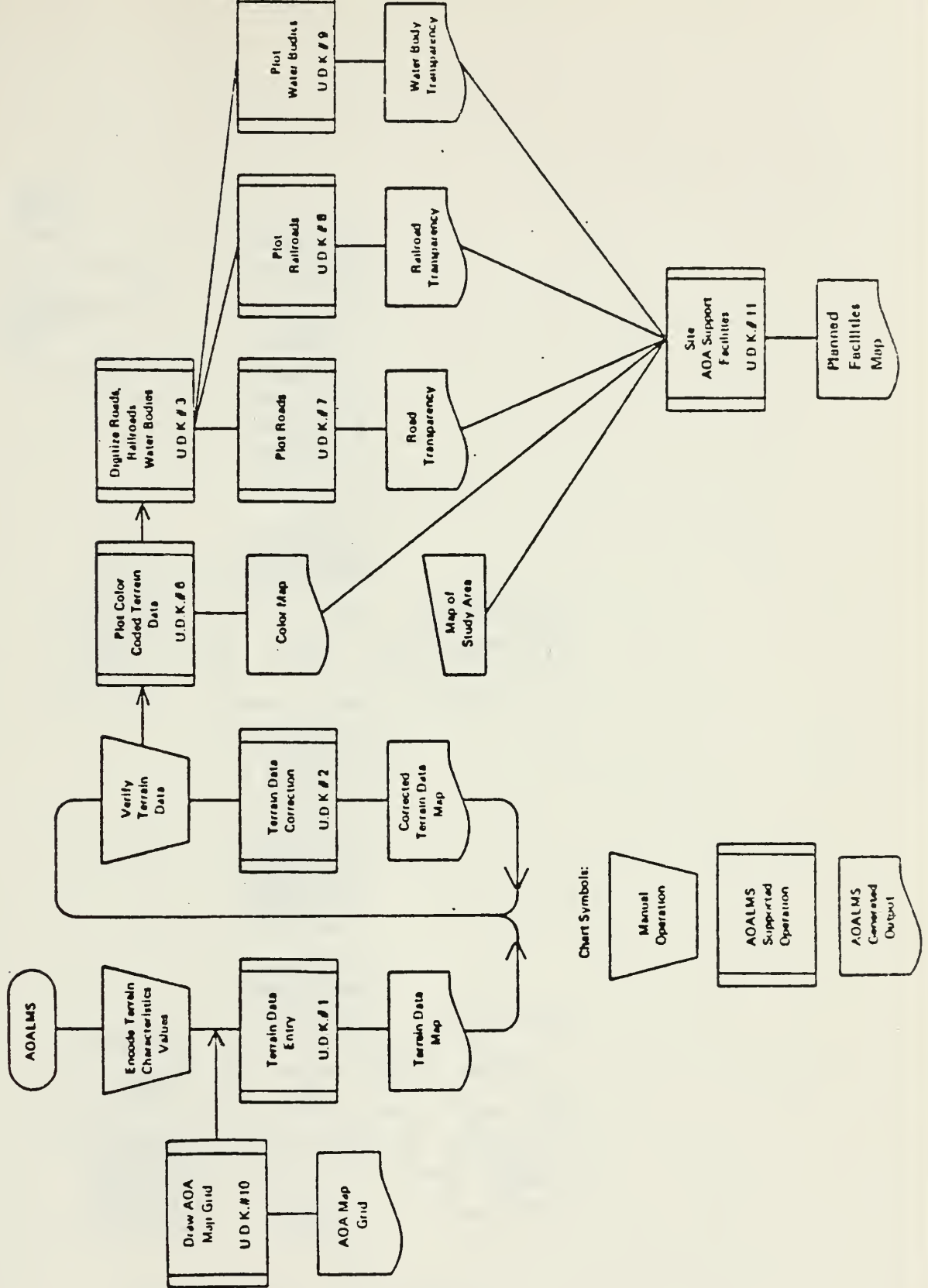
The AOALMS program was developed for the Marine Amphibious Force as a means of optimizing the siting of facilities for an advanced base.¹⁸ The input data is the terrain characteristics (TC) of various sites within the AOA. These sites are divided into 500 m x 500 m square grids. Each square of the grid is then described by the three digits listed below:

1. Vegetation-density of its foliage (barren through jungle), (1=Bare, 2=Grass, 3=Brush, 4=Forest, 5=Jungle Urban).
2. Topology - contour (flat through cliffs), (1=Flat, 2=Rolling, 3=Hilly, 4=Cliffs).
3. Geotechnics - suitability of its soil for construction (good thin through bad), (1=Good thin, 2=Good thick, 3=Bad).

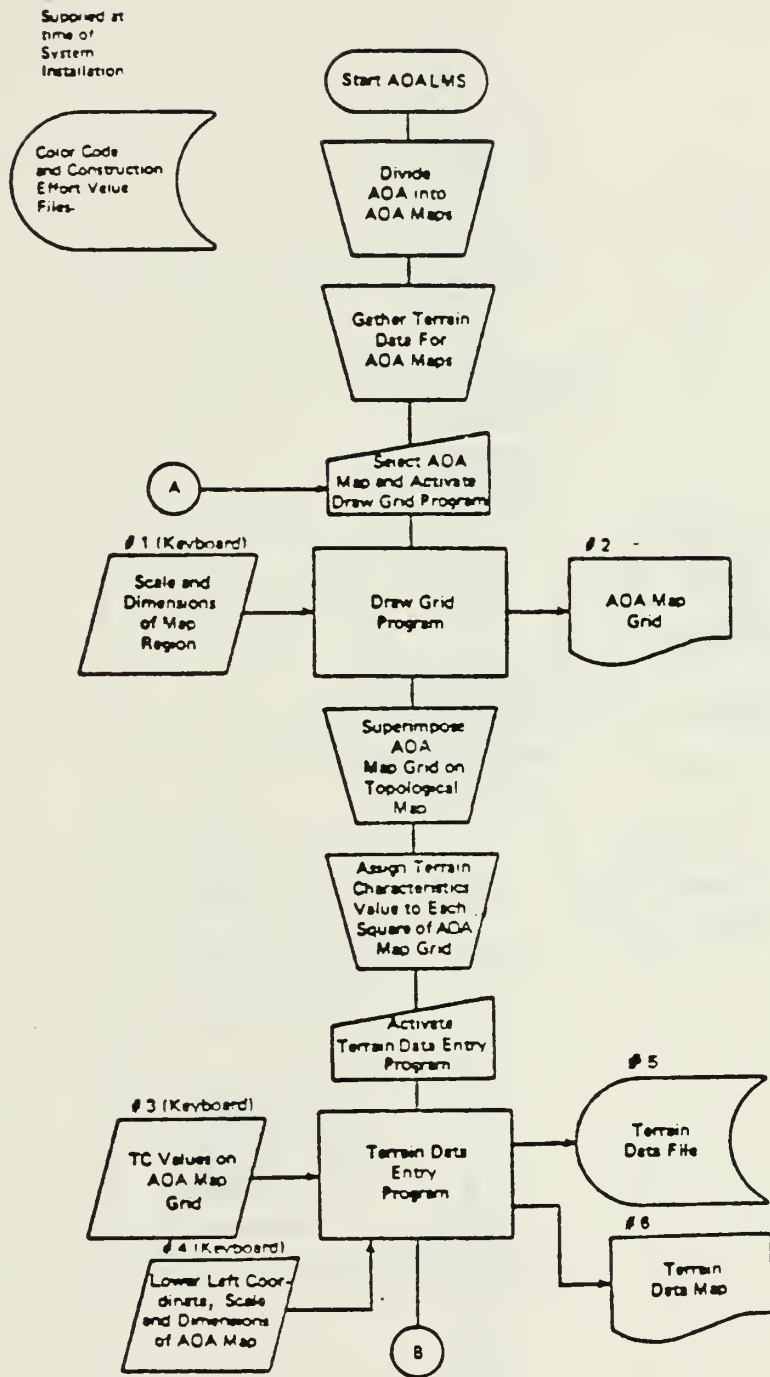
These three digits define the TC of each grid square, and must be manually input to the microcomputer. The AOALMS program evaluates the construction effort (CE) for construction of the various types of facilities based on the TC values for those grid squares. This CE value is the amount of man-days required to complete stripping, clearing, grading, and alike for that particular grid. The various types of facilities that are considered are: Expeditionary Airfield (EAF), Logistic Support Area (LSA), Ammunition Supply Point (ASP), Airfield of Varying Length (AIR), and Petroleum-Oil-Lubricant Facility (POL).

Finally, salient geographical features such as roads, rivers, and railroads are entered into the data base; along with the proposed location of each type of facility. Up to three different proposed locations or orientations is permitted for each facility type. The AOALMS program outputs the construction effort required for each proposed facility site. The user picks the optimal site based on the CE criteria. When all sites have been chosen the program adds up the total construction effort required for the AOA. Figures 4.5 through 4.9¹⁸ define a flow chart of this process.

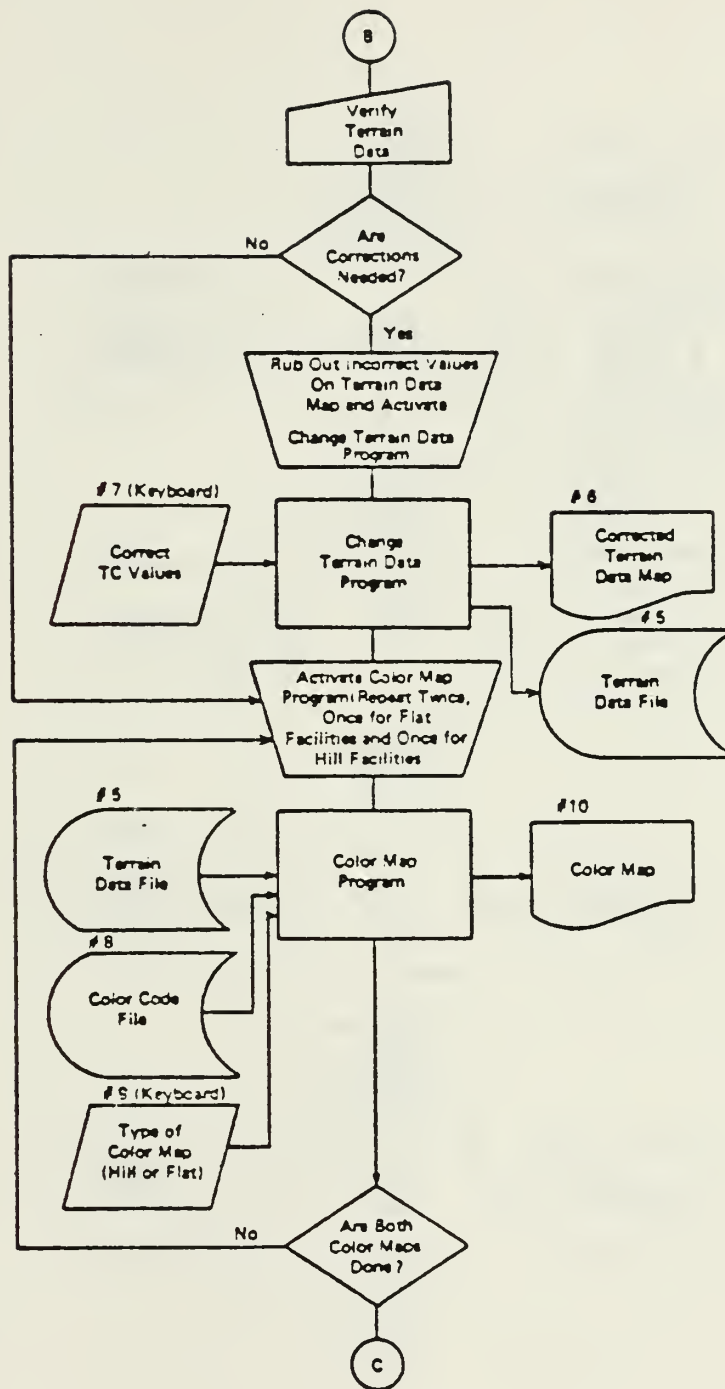
The full Tektronix system is shown by Table 4.10¹⁸. It is felt that a modified version of the Tektronix system would be suitable for both the DEP and ADALM systems. The Marine Corps is expected to procure this system in the near future.



AOALMS System Flow Chart
Figure 4.5

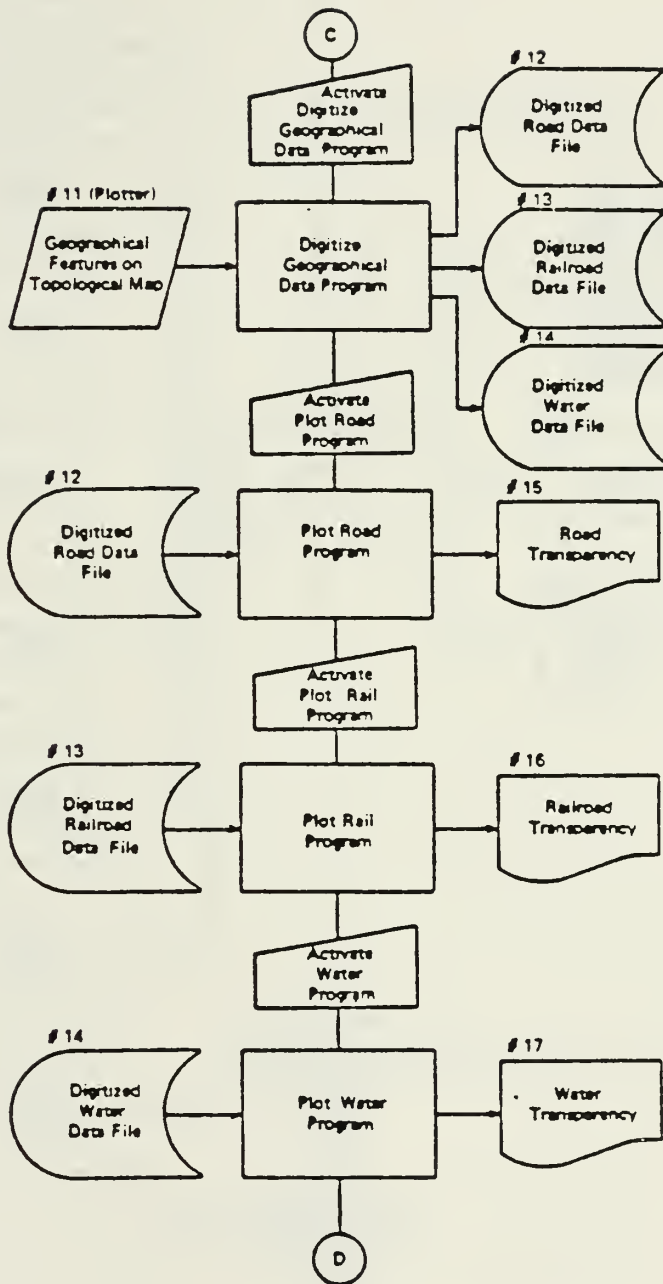


AOALMS System Flow Chart
Figure 4.6



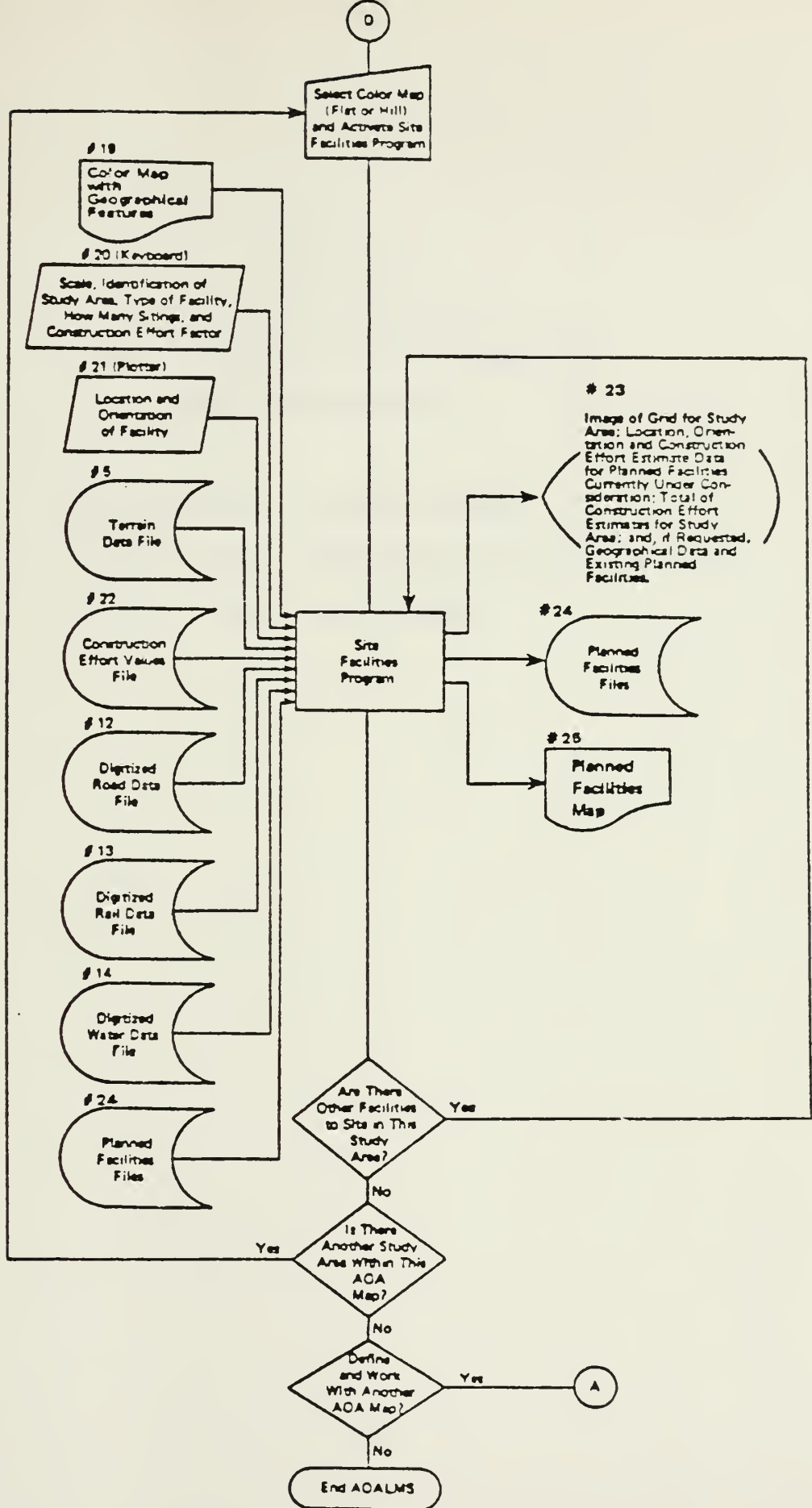
AOALMS System Flow Chart

Figure 4.7



AOALMS System Flow Chart

Figure 4.8



AOALMS System Flow Chart

Figure 4.9

TABLE 4.10

Proposed Computer System Components
For MAF Engineers

- * Tektronix 4052 Graphic System
- * Tektronix 4662 Interactive Digital Plotter
- * Tektronix 4907 File Manager

CHAPTER V

GEORGIA INSTITUTE OF TECHNOLOGY CONSTRUCTION MANAGEMENT PROGRAMS

As previously discussed, many of the prototype programs developed by the Georgia Institute of Technology came about as a result of research performed for the Civil Engineering Laboratory. This contract was not the total source however. A most notable exception is the Water Points program developed by Monty Anderson for his Master's thesis.

In "A Prototype Decision Support System for the Location of Military Water Points"¹⁹ Anderson utilizes operations research techniques to optimize the siting of proposed water re-supply points based on the locations and quantities of both the water supply points and the demand points. Anderson saw this problem as one ideally suited to a microcomputer environment, and developed the program on a Chromatics CA-1999 color graphic microcomputer. His method of input is in itself rather unique as it utilizes both keyed data and graphic interactive input by the use of a light pen. The user initializes data by keying in unit demands and water point capacities. The road network linking the units with the supply points is then generated using a bit pad and digitizer. Finally the light pen is utilized to mark the water point locations and the units to be supported. These are indicated as nodes along the road network.

The user then manipulates the water allocations by making light pen "hits" to the program menu items or points appearing on the color monitor representing supply points or unit locations. In this manner the program defines the shortest distance to each unit from the water points and allocates the water supply to the units so that the distance travelled is minimized.

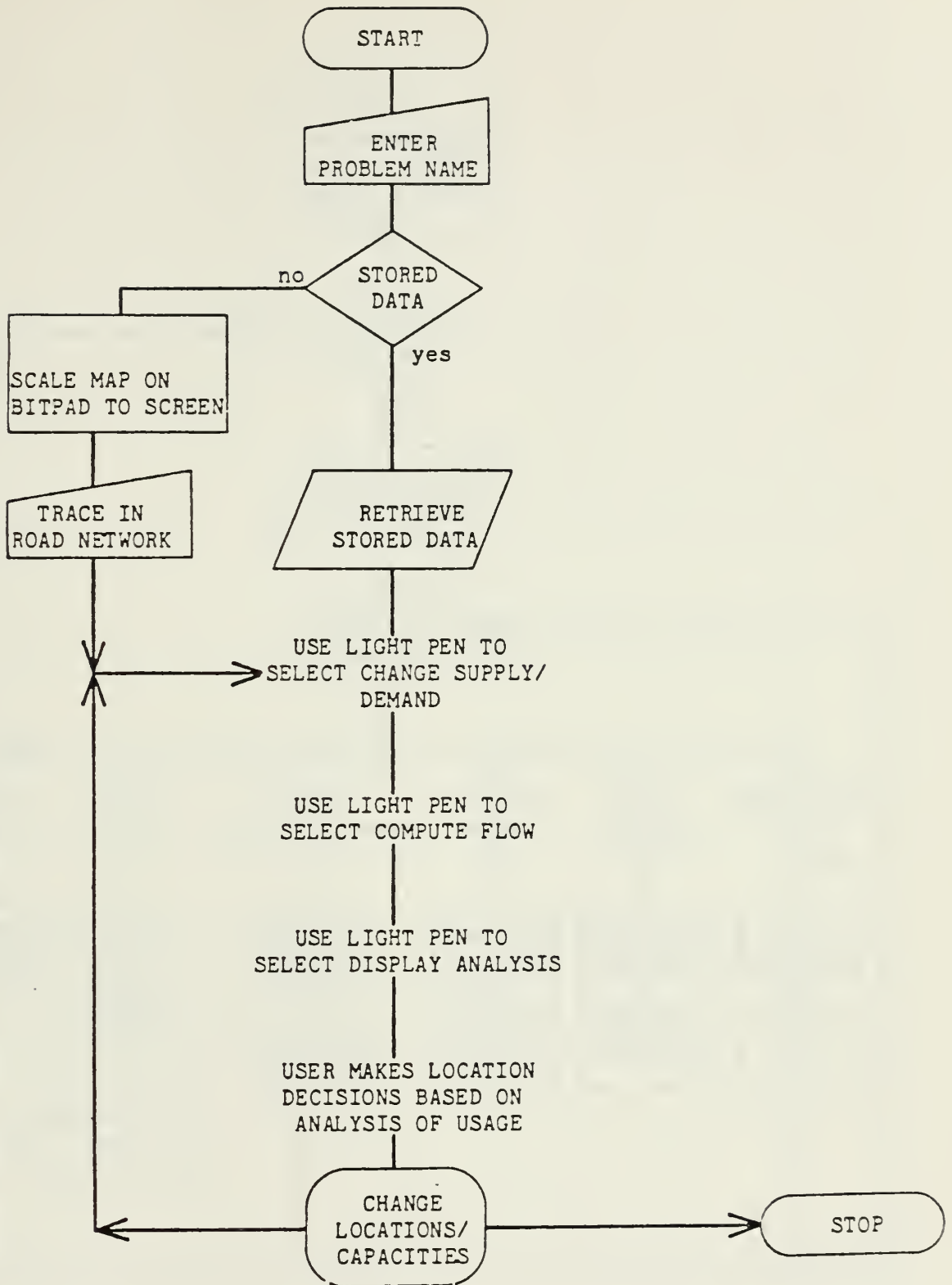
The use of graphical displays in the formulation of the problem and a user manipulated light pen in the problem's solution appears to be an ideal method for interactive user interface. As the user "sees" where his resources are over or under utilized and can change the allocation with a flick of the light pen, analysis of many different distributions can be accomplished in a relatively short time. Even more important, the user can more clearly understand the problem when a graphical display is used.

The system can generate four different graphical reports to aid the user's decision in the reallocation process. They are:

1. Gallons supplied vs Site #
2. Distance travel per gallon vs Site #
3. Actual distance travelled vs Unit #
4. Distance travel vs Alternate Water Point assigned.

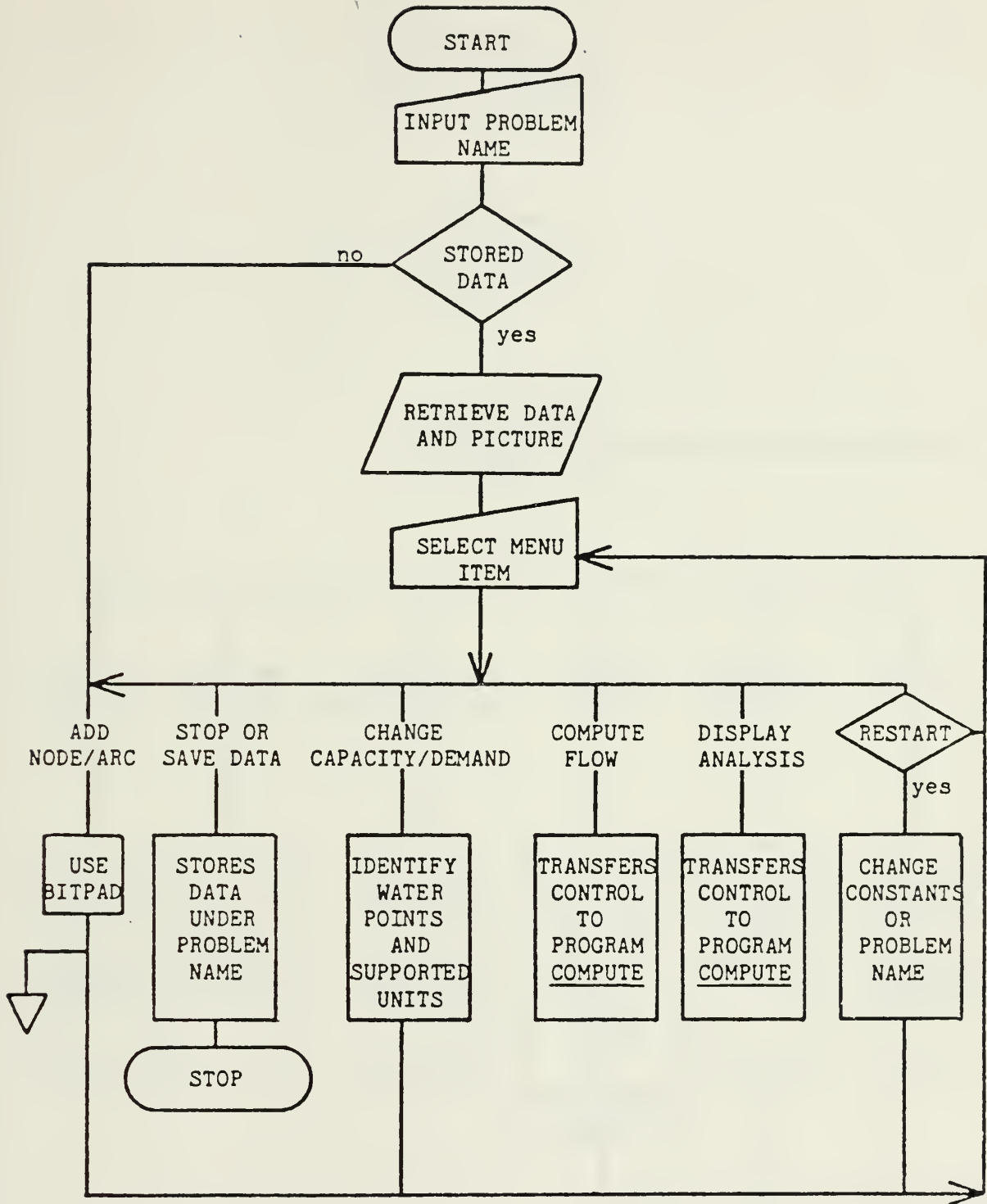
In addition, numerical data is available in each of the four graphical reports. Figure 5.1 shows a flow chart of user activities in manipulating this program. Figures 5.2 and 5.3¹⁹ are flow charts of the program's two main subroutines.

Another notable effort in the development of computer aided



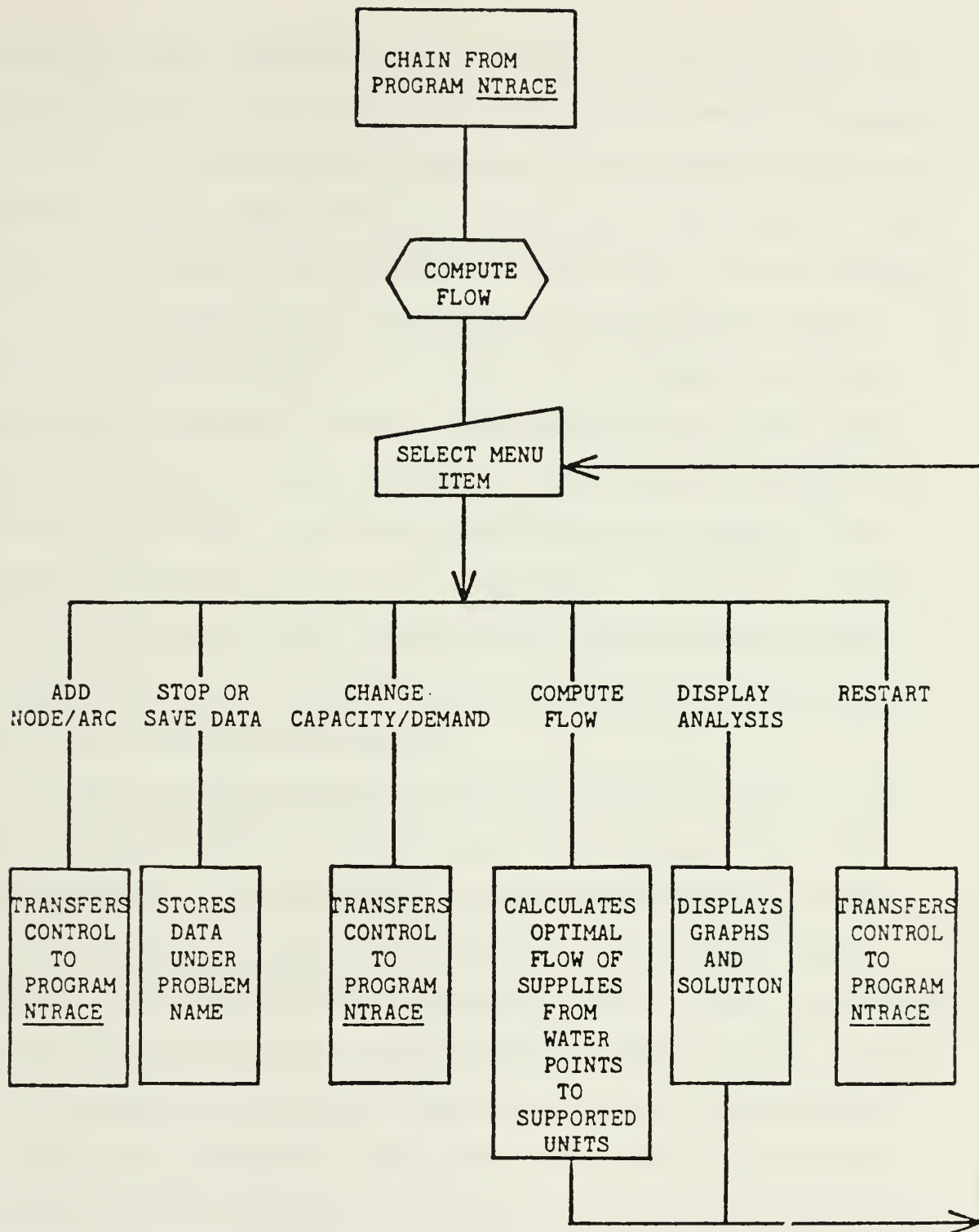
Water Points Flow Chart

Figure 5.1



Subroutine NTRACE Flow Chart

Figure 5.2



Subroutine Computer Flow Chart

Figure 5.3

designs for microcomputers is the special research problem of Dwight S. Durham. In the "Use of Computerized Network Analysis Systems on U.S. Army Corps of Engineers' Construction Projects"²⁰, Durham develops a small precedence notation CPM program. This program was written for interactive input mode. Input consists of activity number, title, following activities, and "work division". Work division is a code used to selectively sort various activities for output. The output report gives early start, late start, early finish, and late finish workdays, as well as total float. Critical activities are flagged. The program can handle up to fifty activities at one time. Durham modeled his program after the CYCLONE * GAME program of reference(21), using selective portions of the HGAME file to produce a simplified networking program.

"Computer Aided Design for MAF Engineers"²² by Dr. D. W. Halpin was the impetus for the many of the recent CAD's for microcomputers. It was written as an investigation into the existing level of development of MAF suitable design support systems. Civil Engineering Laboratory, Port Hueneme, commissioned this study as part of its research for the DEP system. As part of this study two of the MAF engineering tasks were developed into prototype programs. The tasks chosen were prefabricated floating bridging design, and helicopter landing site and zone preparation design.

The Prefabricated Floating Bridge program was based on design procedures described in the Army River Crossings and Bridging Manuals (Field Manual 90-13 and Technical Manual 5-210).

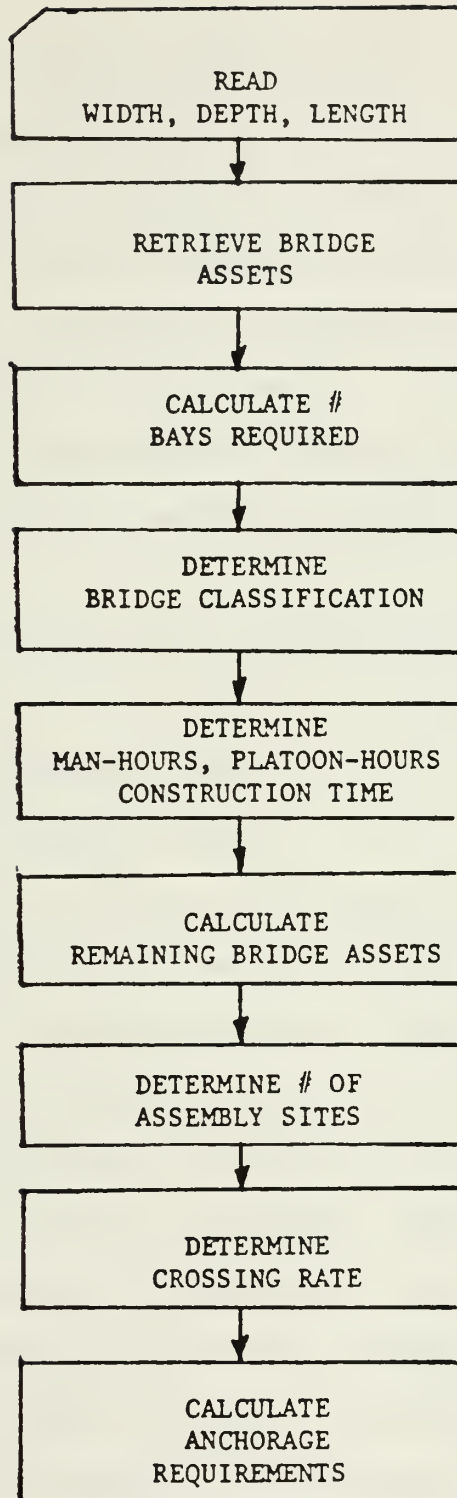
The design is for the most common types of floating bridges: Mobile Assault Bridge, Ribbon Bridge, and M4T6 Bridge. Input to the program is interactive with the user. Typical input information is as follows:

1. Availability of bridging assets
2. River width, depth, and current velocity
3. River bank height, slope, and soil CBR
4. Type of bridge to be built (one or any combination of bridge types)
5. Conditions under which the bridge is to be built (night or day).

The program then evaluates the site for design. This evaluation includes a feasibility study of the type of bridge to be used, with the program offering alternate methods of river crossing if appropriate. With valid input, the programs output report provides the following information:

1. River data (width, depth, and velocity)
2. Bridge classification (wheeled and tracked)
3. Crossing rate (vehicles per hour)
4. Bridge construction time (Platoon and/or Company Hours)
5. Required number of assembly sites
6. Required bridge assets
7. Remaining bridge assets
8. Any special requirements
9. Anchorage requirements

Figure 5.4²² shows a macro flow chart of this program.



Prefabricated Floating Bridge Program Flow Chart

Figure 5.4

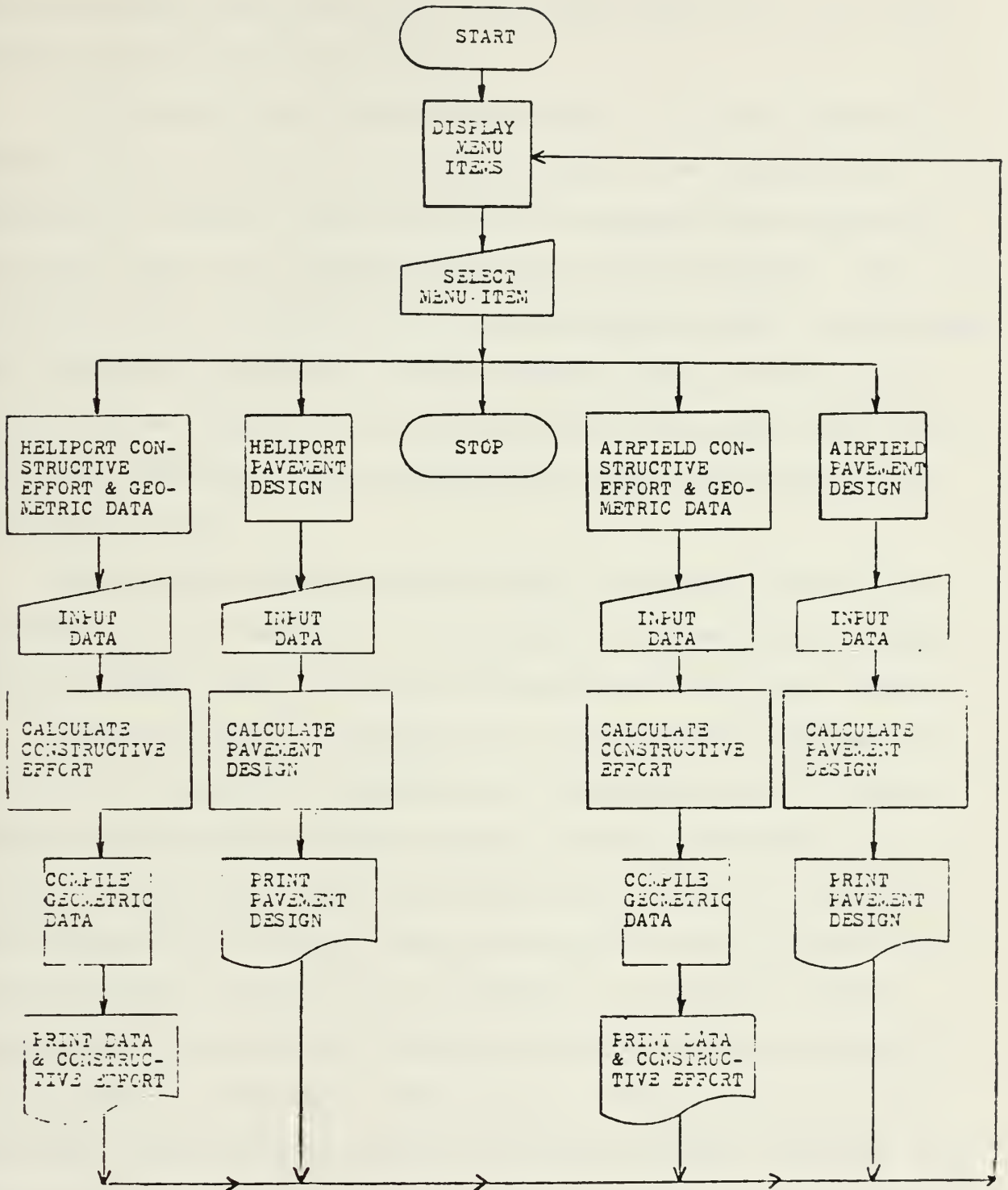
The helicopter landing site program developed under the Computer Aided Design for MAF Engineers study was later expanded into a special research project. In "Computer-Aided Design of Heliports and Expeditionary Airfields in the Amphibious Operations Area",⁴ Hambrock develops a prototype program for the selection of an optimum airfield site. In addition, the program provides final layout and pavement design data.

The design of this program was divided into four basic functions:

1. Heliport Constructive Effort and Geometric Data
2. Airfield Constructive Effort and Geometric Data
3. Heliport Pavement Design
4. Airfield Pavement Design

This is described by the macro flow chart of Figure 5.5.⁴ Input information is gathered by interactive input with the user. Input common to all functions include: the site's soil type, soil condition, soil thickness, vegetation, terrain slope characteristics, expected number of cycles of aircraft/helicopter operations, airfield index of subgrade, CBR of subgrade, CBR of base, CBR of subbase (if applicable), and construction force employed. In the case of a heliport design, additional information such as the number and type of helicopters, and the configuration of the field need to be inputted. The airfield design requires temperature, runway gradient, elevation of runway, and elevation adjustment factor data to be inputted for the runway length calculations.

The program's design of the runway is based on the U.S.



Helicopter Landing Site Program Flow Chart
Figure 5.5

Army Technical Manual TM5-330,^{2 3} and LCDR Hambrock's own field experience with airfield design.

The output is in a narrative format. Only the geometric design of the airfield is in tabular form. For the Heliport designs the typical output would be the landing pad dimensions, shoulder sizes, and approach/departure zone dimensions. The airfield output consists of the runway length, width, longitudinal and transverse gradients; taxiway length, width and gradients; parking apron length, width and gradients; clearance zones; over-run area dimensions; approach/departure zone requirements; and separation zones.

The pavement design output format is common to both heliport and airfield designs. It consists of pavement type, thickness of design layers, and basic quantities of materials needed.

Another program developed for use by the MAF engineer is part of the "Equipment Estimating and Scheduling for the Marine Amphibious Force"^{2 4} special research project, developed by Lt. D. Pruett. MAFESS (MAF Equipment Scheduling System) is a scheduling program which utilizes basic network input and equipment requirements for each activity, to develop a "crashed" network governed by the duration of these activities and the availability of equipment on each day. The program utilizes the Bennett algorithm to perform the crashing mechanism and Rood equipment selection cards to estimate the productivity of the various equipment alternatives. MAFESS is a prototype program based on modified version of FLEET, a computer program developed by R. Morrison at the Georgia Institute of Technology.

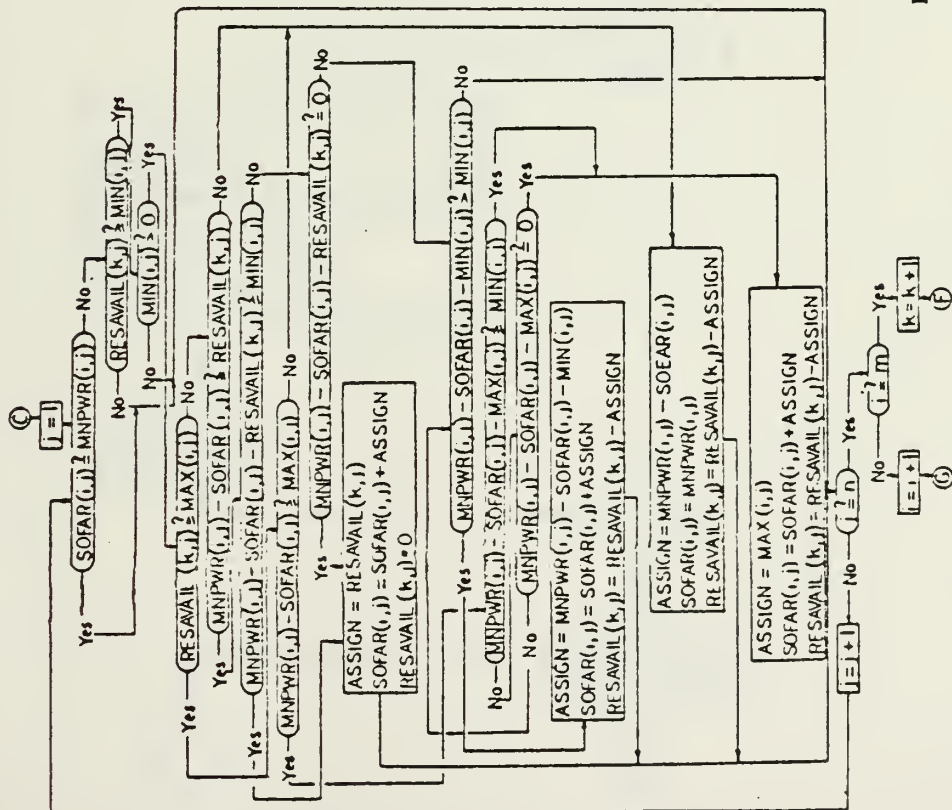
The input is in the interactive format, with the user describing the quantity and conditions under which the work activities are to be performed. The ROOD equipment card data base converts this information into required "resource-days" to complete the activity. The EQUALOC subroutine of the MAFESS program uses the Bennett algorithm to assign the equipment to the activity network, "crashing" the network where necessary to schedule equipment constrained activities. The final output report assigns each type of equipment on a day by day basis to each work activity requiring this equipment. Figures 5.6 and 5.7²⁵ shows the flow chart of the EQUALOC subroutine.

The state of the art, in relation to this problem, is the special research project of LCDR Robert Setzekorn. Setzekorn saw the design of an improved road in the AOA as a task ideally suited to microcomputer processing and solution. To ensure that the memory storage capacity of the microcomputer chosen to implement this program was not exceeded, Setzekorn divided his program into a software package with five functional stages. Those stages are Horizontal Alignment, Superelevation, Vertical Alignment, Soils, and Cross Section. Figures 5.8 and 5.9¹ show the flow chart of the program and the relationships between the five functional stages.

The overall program was designed for interactive input. The user is required to provide a rough layout of the road on a grid map. From this layout the user is asked to provide input information. Typical input requirements would be:

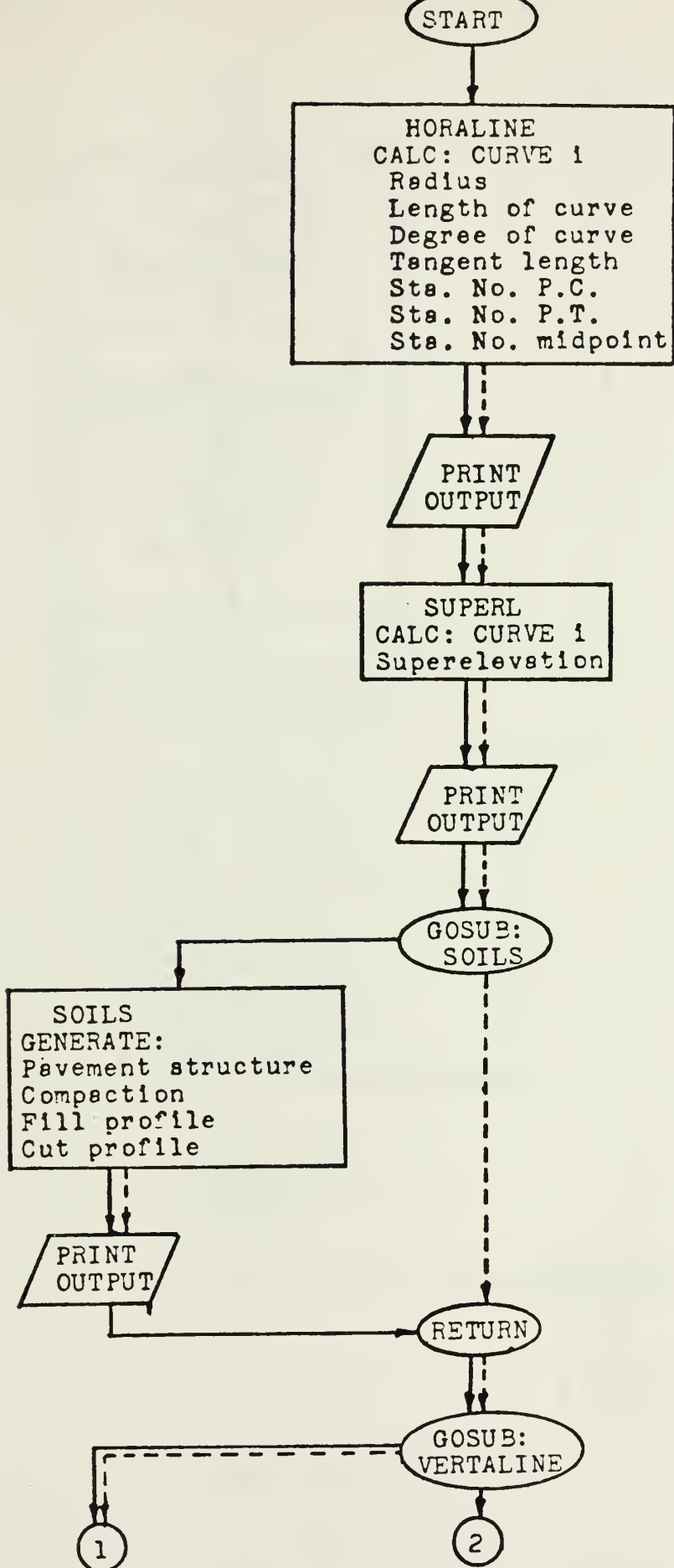
1. Number of curves

ASSIGN = resource assignment;
I = activity number = 1, 2, . . . m;
IBEGIN (I) = begin event number for activity I;
IDUM (I) = dummy designator for activity I { 0: Real activity, 1: Dummy activity;
INUM = counter used with **ISAME** to determine whether all activities entering event **NUM** have been completed;
IRES = counter used to determine whether all of an activity's required resources have been assigned;
ISAME = counter used with **INUM** to determine whether all activities entering event **NUM** have been completed;
ISTAT (I) = status designator for activity I { 0: Not yet eligible for scheduling, 1: Eligible for scheduling, 2: Resource scheduling underway but not completed, 3: Resource scheduling completed;
ISTAT 3 = counter used to determine whether all activities have been scheduled;
J = resource number = 1, 2, . . . n;
JEND (I) = end event number for activity I;
k = day number = 1, 2, . . . 400;
m = number of activities;
MAXEVENT = maximum event number;
MAX (I, J) = maximum amount of resource J that can be assigned to activity I;
MIN (I, J) = minimum amount of resource J that can be assigned to activity I, if any is assigned;
MNPWR (I, J) = total requirement of resource J for activity I;
n = number of resources considered;
NUM = counter on begin and end event numbers, used to determine whether the entire network has been searched;
RESAVAIL (K, J) = amount of resource J available on day K; and
SOFAR (I, J) = amount of resource J that has been assigned to activity I at any given time.



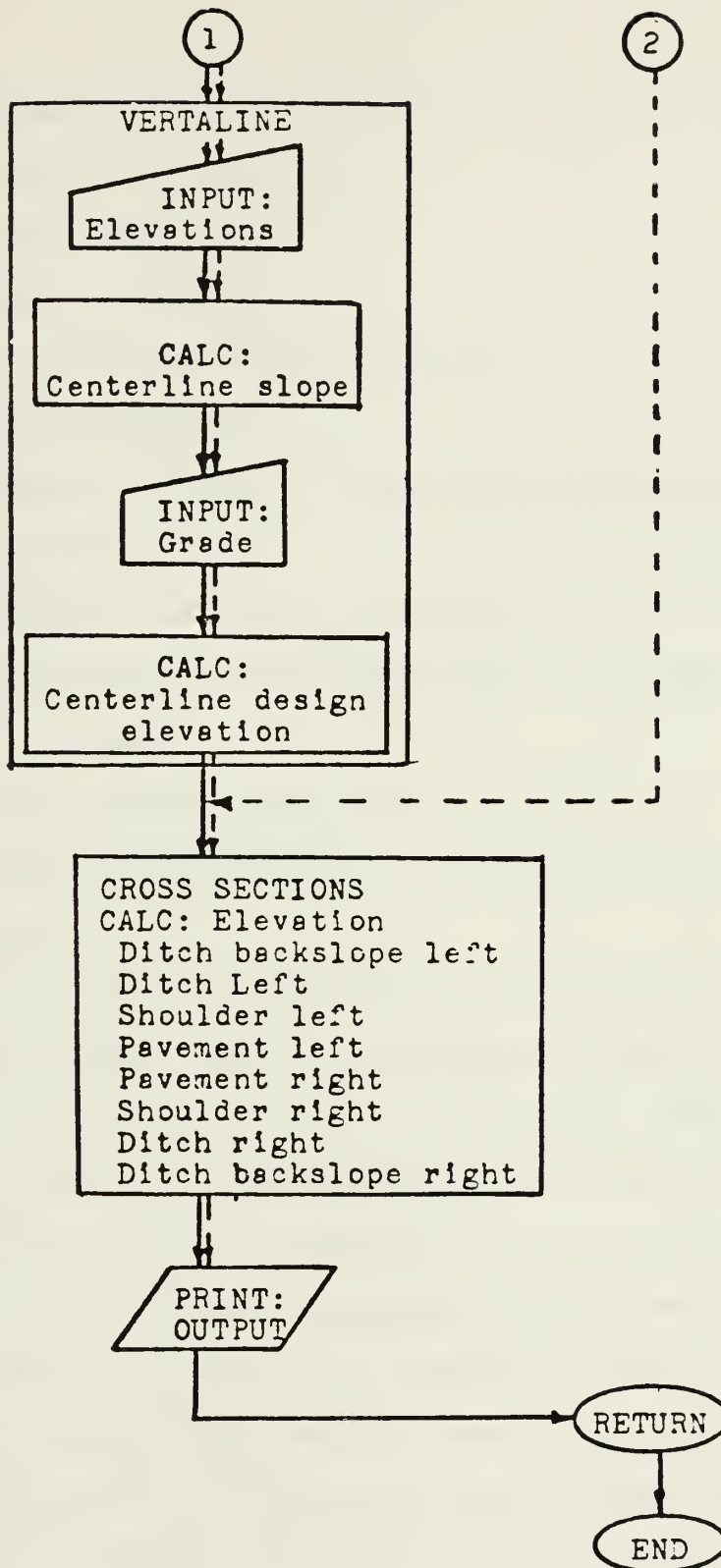
EQALOC Subroutine Flow Chart

Figure 5.7



Road Design Flow Chart

Figure 5.8



Road Design Flow Chart

Figure 5.9

2. Centerline intersection of each curve
3. Angle of intersection
4. Chord length
5. Design Velocity
6. In-place density of subgrade
7. CBR of subgrade
8. Numbers and types of base and subbase materials and CBR of each
9. Number of stations in section
10. Elevations of centerline, left and right side of pavement
11. Grade between stations
12. Number of stations to be included in a vertical curve.

The Horizontal Alignment and Superelevation subroutines develop station data based on the length of the curve. All other subroutines provide calculation based on 100 foot stations. However the user is not necessarily constrained by this interval. The output data is both in graphical and tabular form. The pavement and base design is graphical, where the distance between layers on the cross section is indicative of its relative thickness. The horizontal and vertical curve data, and the superelevation information is given in tabular form. Both Setzekorn's program and the program developed in one's research assume the user is an engineer with highway/surveying background. The need for this background will be discussed in the next section.

CHAPTER VI

PROGRAM DESIGN

The design methods used in the development of this program are, for the most part, procedures developed in Army technical manuals.^{26, 27, 28} The program was developed in eight functional stages; the first five of which were derived from the special research of LCDR Robert Setzekorn. These eight functional stages are listed below:

1. Horizontal Alignment
2. Superelevation
3. Soils
4. Vertical Alignment
5. Cross Section
6. Intersection Points
7. Cross Sectional Area
8. Mass Balance

The last three stages and the Vertical Alignment stage rely to some degree on the design procedures expressed by references (29) and (30). These design procedures will be fully explained later in this chapter. The overall program, FIRD (Fortran Interactive Road Design) was developed as a series of subroutines which handle the eight functional stages.

As the program was written for interactive input, care must be taken to insure the proper format of input data. To this purpose a user's manual for the program was prepared, and is listed as Appendix C. Similarly, a detailed flow chart is listed in Appendix D. It may be useful to refer to these

appendices during the below discussion of the design procedures.

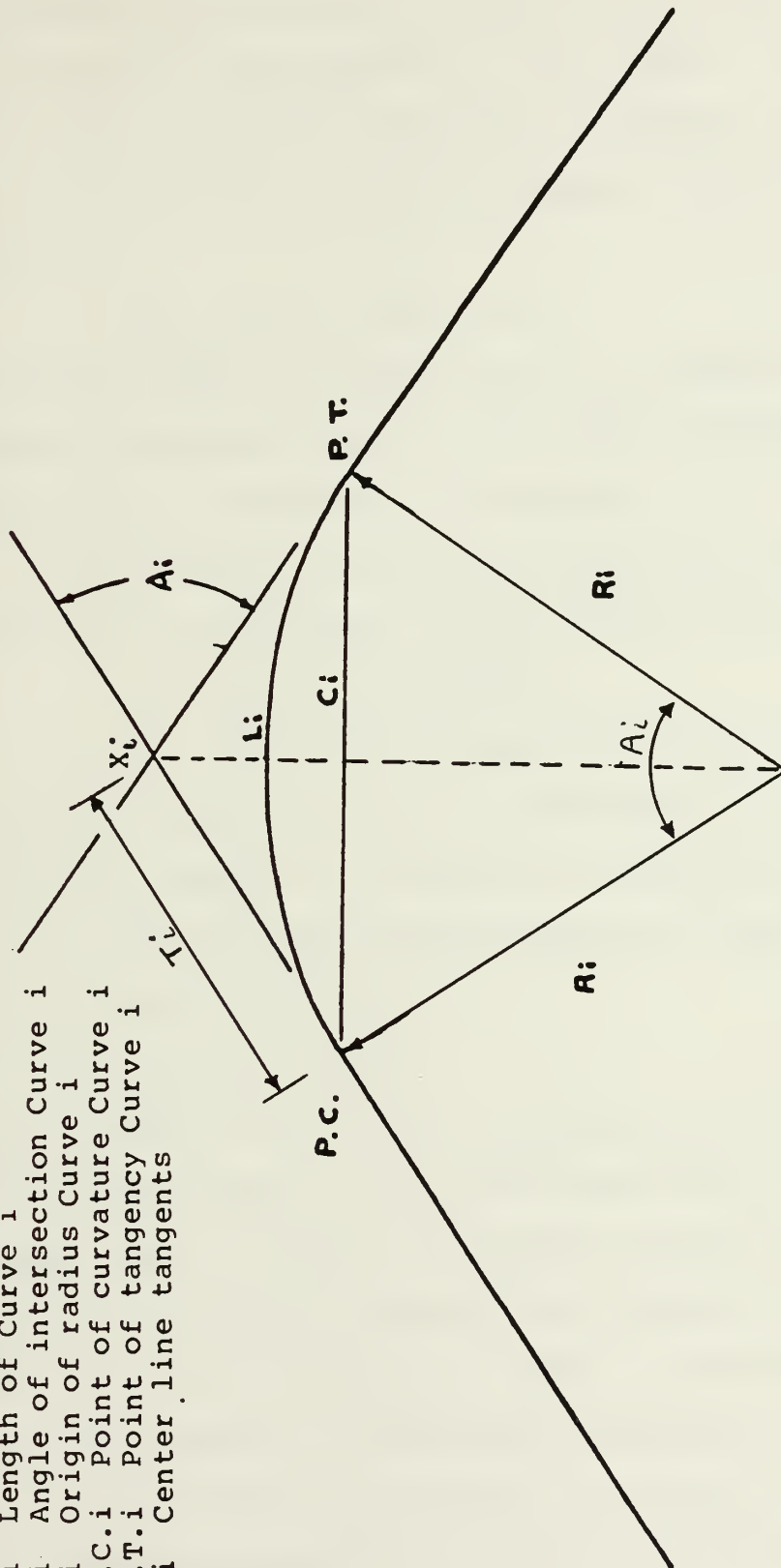
Horizontal Alignment

The first design item to be addressed in any road design is the horizontal alignment. Here the engineer selects the alignment of the road centerline. Usually this is accomplished by studying contour or terrain maps of the area in question. The engineer attempts to select that route that most expeditiously reaches the desired location, while minimizing cuts and fills and avoiding natural obstacles. Layout is usually restricted to a series of straight line segments and curves. This alignment is recorded by stations, where straight line segments are assigned full station numbers at 100 foot intervals. Rarely will the beginning point of a curve (PC) lie at a full station.

The horizontal curve programming of FIRD was designed to handle only simple curves. Complex curves can be approximated however by combining a series of adjoining simple curves. An example of a simple curve and curve component definitions are given by Figure 6.1.¹ The program starts interactive input by requesting the number of horizontal curves in the road section being designed. Up to twenty horizontal curves can be designed per road section. If no curves are to be designed in the road section, the user types in \emptyset . In this case the program bypasses the Horizontal Alignment and Superelevation sections and proceeds directly to the Soils section.

If horizontal curve data is desired, the computer requests the station number of the point of intersection of the tangent lines. This point is shown by point X on Figure 6.1. Next

- Ri Radius of curvature-Curve i
- Ci Chord length of Curve i
- Li Length of Curve i
- Ai Angle of intersection Curve i
- Oi Origin of radius Curve i
- P.C.i Point of curvature Curve i
- P.T.i Point of tangency Curve i
- Ti Center line tangents



HORIZONTAL CURVE GEOMETRY

Figure 6.1¹

the angle of intersection between the two tangent lines (A_i), and the curve chord length (C_i) are input. Once these have been input the program proceeds to calculate the curve data.

The first calculation to be made is the Radius of Curvature (R):

$$R_i = C_i / (2 \times \sin (A_i / 2)) \quad 6.2$$

Though the angle of intersection A_i is inputted in degrees, the program internally converts this angle to radians prior to the calculation of horizontal curve data.

The length of the curve is calculated next. This is defined as the arc length of the inscribed circle:

$$L = A_i \times R_i \times 3.14 / 180 \quad 6.3$$

The calculation of the degree of curvature is next considered. The program limits this value to be less than ten for safety reasons. The degree of curvature (D_i) is defined as the angle of intersection between the radii of an arc whose chord length is 100 feet.

$$D_i = 5729.28 / R_i \quad 6.4$$

If the degree of curvature is determined to be more than ten, the program warns the user and returns him to the point in the program which requests the station number intersection of the tangent lines (X_i). The user may decrease the degree of curvature value by decreasing the angle of intersection, increasing the chord length, or completely realigning the curve.

If the degree of curvature is found to be less than ten, the program proceeds to calculate the tangent distance (T_i) from the point of intersection to point of curvature (P.C) and

point of tangency (P.T.).

$$T_i = R_i \tan (A_i/2) \quad 6.5$$

From this calculation the station numbers of the P.C., P.T., and midpoint of the arc length (M_i) are determined:

$$P.C. = X_i - T_i \quad 6.6$$

$$M_i = P.C. + L/2 \quad 6.7$$

$$P.T. = P.C. + L \quad 6.8$$

Finally, the program outputs all the horizontal curve data in tabular form. This output consists of the following:

1. Station Number of the Midpoint of the Curve
2. Radius of Curvature
3. Length of Curve
4. Degree of Curvature
5. Tangent Length
6. Station Number of the P.C.
7. Station Number of the P.T.

Though the horizontal curve design stage has been completed, a redesign of this curve may be necessitated by the safety features of the superelevation design stage. This will be discussed later in this chapter.

Superelevation

Superelevation of a road section is used by the engineer to prevent slippage of vehicle tires while a vehicle is moving along a curve. It is a safety feature most commonly used when the degree of curvature (D) and designed road speed (V) are high.

In this program, the user has the option of four design road speeds: 30, 40, 50, or 60 miles per hour. If a design road speed other than these are selected, the user is instructed to reselect.

If a valid velocity has been selected, the program calculates a superelevation ratio, using the equation listed below:

$$E = (V^2/15 \times R) - F \quad 6.9$$

In this equation, V is the design speed (in miles per hour), R is the radius of curvature (in feet), and F is a side friction factor. The side friction factor is a function of the design speed and is stored within the program as follows:

<u>V (mph)</u>	<u>F</u>
30	0.16
40	0.15
50	0.14
60	0.13

The program automatically chooses that side friction factor that corresponds with the input design speed.

If the calculated superelevation is zero or less, the program assumes no superelevation on that road section. If the superelevation ratio is between zero and 0.08 foot per foot, the program accepts this value and stores it for output. If the superelevation ratio is between 0.08 and 0.10 the program asks the user to input whether snow is prevalent in the area. If snow is not a factor, the program accepts the superelevation value and continues on. If snow is prevalent, the program warns the user that the superelevation ratio is too high and returns the user to the beginning of the horizontal alignment stage for curve redesign. In any case, if the superelevation ratio exceeds 0.10 the program will automatically call for a horizontal curve redesign.

As reinput of all curve data is necessary in the event of excessive superelevation values, it is important for the engineer to carefully evaluate this condition before initializing the curve data. Sometimes a series of transition curves can be used to provide a gradual cross section change.

If the calculated superelevation ratio is acceptable, the program proceeds to output all the horizontal curve and superelevation data in tabular form. This output consists of the following:

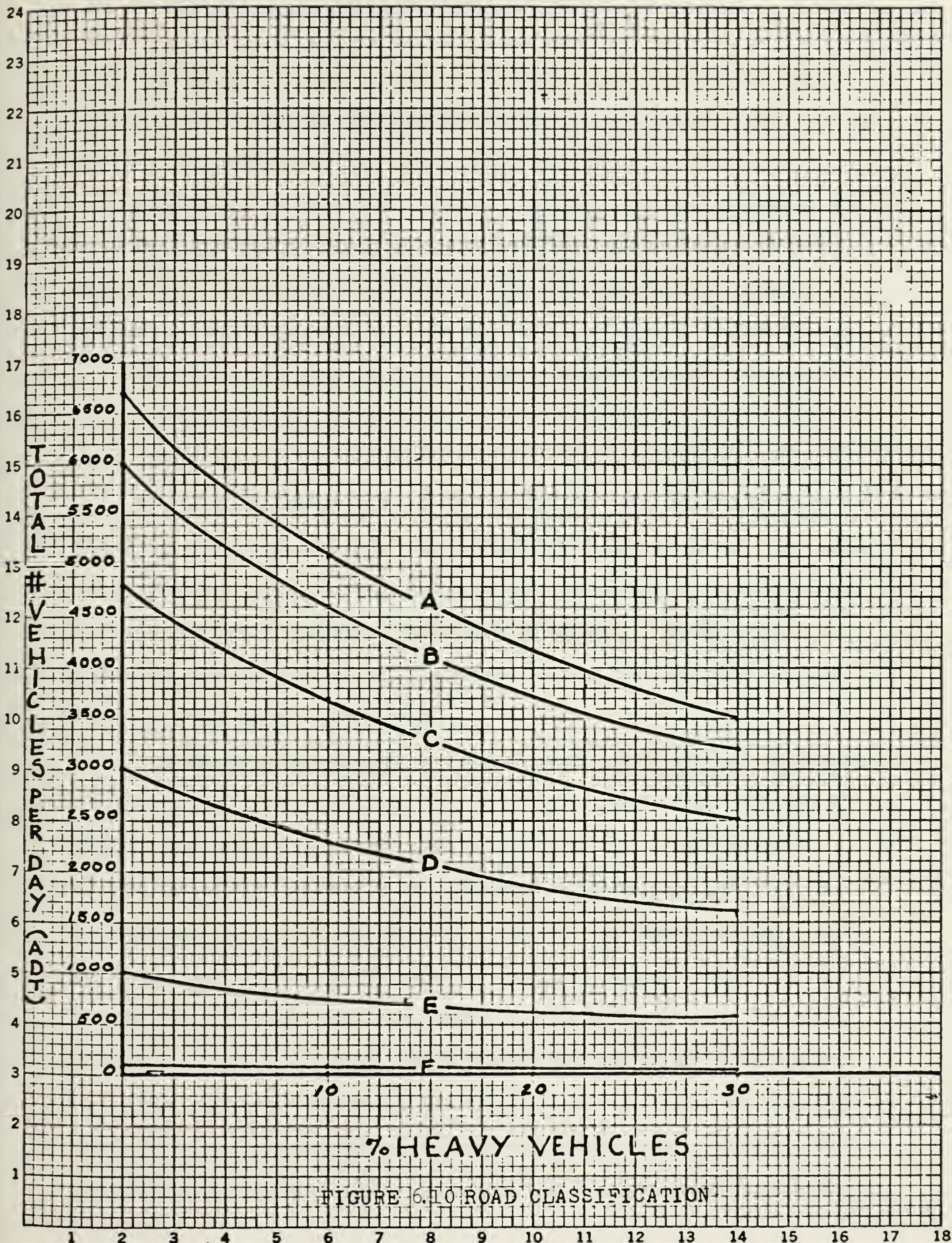
1. Station Number of the Midpoint of the Curve
2. Radius of Curvature
3. Length of Curve
4. Superelevation
5. Tangent Length
6. Station Number of the P.C.
7. Station Number of the P.T.

Some of this data is stored for later use in the Cross Section Design stage of the program.

Soil Design

The next design stage, Soils, designs the road section structurally, such that it is able to accept the designed loads. First, however, it will be necessary for the engineer to determine what amount and types of traffic will utilize the road. This entails an estimate of the Average Daily Traffic (ADT) and the percentage of that ADT comprised of vehicles heavier than passenger autos and light pickup trucks. By Figure 6.10, the engineer is able to classify the road by the ADT and the percentage of heavy vehicles.

For example, suppose the ADT is expected to be 3300 vehicles per day and the percentage of heavy vehicles is estimated at



7% HEAVY VEHICLES

FIGURE 6.10 ROAD CLASSIFICATION

22%. By plotting this point on Figure 6.10, it may be seen that the curve above that point is B. Therefore a class B road would be required.

Similarly Figure 6.11 allows the engineer to convert the road classification into a design index. By taking the road classification B and a twenty-two percent of heavy vehicle estimate into Figure 6.11, it can be seen that a design index of 5 would be required. Both these design indices are based on a heavy vehicle classification of 18,000-pound equivalent axle loads.

The design index will be used by the program to design the flexible pavement and the subsurface base structures. As with manual design procedures, this is accomplished by determining the total road structure above the subgrade. From this calculation the design of the pavement, from the surface to the subbase, can be accomplished.

The user first inputs a road design index and an estimate of the actual in-place density of the subgrade into the program. The subgrade density must be input as a percentage of the modified proctor. The program then requests input on the number of soil layers to be used in the substructure (base, subbase, and select materials). Usually no more than three different materials are necessary. The program has the capability of handling up to seven different materials, however.

The surface thickness is established by input of the CBR (California Bearing Ratio) of the selected base course. The program accomplishes this by selecting the pavement thickness

% HEAVY VEHICLES

		0	10	20	30
ROAD CLASSIFICATION	A	3	4	5	6
	B	3	4	5	6
	C	3	4	4	6
	D	2	3	4	5
	E	1	2	3	4
	F	1	1	2	<u>3</u>

Design Index

Figure 6.11¹

required at that base CBR for the road classification chosen. The below table of data is stored in the program for this purpose:

	A	B	C	D	E	F
CBR \geq 100	3.5	3	2.5	2	1.5	1
\geq 80	4	3.5	3	2.5	2	1.5
\geq 50	5	4.5	4	3.5	3	2.5

After the program selects the pavement thickness, the program evaluates the thickness of the material above the sub-base. This depends upon whether the lower material has a CBR greater than or less than six, and the design index. If the CBR is less than 6, the design index (D1) is incorporated by the following equations:

$$\text{Thickness} = \text{Integer} (1/A1 + (B1 \times \text{CBR})) + 0.5 \quad 6.12$$

$$\text{where: } A1 = 1/(33.4189 + 6.9253 \times D1) \quad 6.13$$

$$B1 = 1/(66.5698 + 14.6764 \times D1) \quad 6.14$$

If the subgrade CBR turns out to be greater than 6, another set of equations are applied:

$$\text{Thickness} = \text{Integer} (1/(A + (B \times \text{CBR})) + 0.5 \quad 6.15$$

$$\text{where: } A = 1/(26.7561 + 4.90148 \times D1) \quad 6.16$$

$$B = 1/(73.9301 + 16.9388 \times D1) \quad 6.17$$

Once the program has calculated the thickness of the surface and base course, it outputs that information and is ready to continue design. However, if the surface thickness required is only one inch or less with a base CBR of 80 to 100 the pro-

gram will recommend multiple bituminous surface treatment. Similarly if the required surface thickness is only 0.5 inches and the base CBR is 100, a bituminous surface is recommended. The overall thickness of the surface and base course is restricted to be a minimum of six inches, with a base course thickness of at least four inches.

The thickness of the subbase and select materials is considered next. The total thickness of the subbase material is equal to the difference between the total pavement thickness and the total surface / base course thickness. From this quantity, the minimum thickness of each of the subbase/select materials chosen is subtracted until the subgrade is reached. For instance if the total pavement thickness is fourteen inches and the surface/base course material total thickness is seven inches, the total subbase material thickness would be $14 - 7 = 7$ inches.

Upon completion of the pavement substructure design, the program outputs a road cross section indicating asphalt pavement, base, subbase, and select material with their associated depths in inches and soil CBR. Then the compaction requirements of the materials are considered.

The program first evaluates each of the selected materials with regard to their CBR values. If the CBR of any of these materials is less than twenty and the subgrade is noncohesive, the compaction of that material is set at 95%. If the CBR of that material is less than twenty and the subgrade is cohesive, compaction is set at 90%. If the CBR of the material is above twenty, compaction is established at 100%. This data is then

output to the user is cross section format with cumulative depth, percent compaction, and the CBR of the material indicated. The subgrade is temporarily assigned a compaction of 95% for noncohesive materials or 90% compaction for cohesive materials.

The program next evaluates the compaction requirements of the fill sections. The thickness of the required fill section for a pavement with a cohesive subgrade is determined by:

$$\text{Thickness} = A1 + B1 \times I \quad 6.18$$

$$\text{where: } A1 = 70.7932 + 14.3841 \times D1 \quad 6.19$$

$$B1 = -0.677999 - 0.136691 \times D1 \quad 6.20$$

Similarly, the thickness of the required fill section for a pavement with a noncohesive subgrade is determined by:

$$\text{Thickness} = A2 + B2 \times I \quad 6.21$$

$$\text{where: } A2 = 119.487 + 25.3872 \times D1 \quad 6.22$$

$$B2 = -1.13 + 0.240128 \times D1 \quad 6.23$$

The variable I in equation 6.18 and 6.21 is a compaction rate. This rate is incremented from 100 to 80 percent in five percent intervals.

If the compaction computed for the fill exceeds eighty percent, or the design thickness of the pavement structure equals the minimum thickness calculated for eighty percent subgrade compaction; a fill utilizing the subgrade material is specified. If the fill compaction is less than eighty percent, the program adds four inches of eighty percent compact subgrade material to the pavement substructure, with the re-

mainder of the fill to be subgrade material at the compaction calculated for the fill. If the compaction of the last select material exceeds the subgrade compaction by more than ten percent, the program has the subgrade scarified to six inches and recompacted, increasing the subgrade material compaction by five percent.

The final section of the soils design stage is the compaction requirements for cut sections. As with the fill soil design, the program compares the thickness required above the subgrade with the designed pavement substructure at varying compactive efforts. These compactions start at one hundred percent and decrease through eighty percent at five percent increments. If at some point, the required thickness is greater than the substructure thickness, the program requires the subgrade to be removed. This subgrade is then replaced at the compaction specified for that minimum thickness and to a depth such that the minimum thickness is achieved. This then becomes the new design thickness and the comparison is repeated. When the in-place density equals or exceeds the specified compaction or the compaction of the replaced material equals eighty percent, the replacement process is stopped. The new subgrade is then scarified to a depth of six inches and compacted to the density of the replaced material immediately above it.

The data for both the fill and cut section pavement structure is output in cross section format. All the cross section formatted outputs within the soils subroutine are keyed such that approximately one-inch of material depth is spaced

four printer lines down.

Vertical Alignment

The vertical alignment design stage requires the greatest interactive effort with the user. The design procedures utilized were developed using the parabola method of reference (30). This differs from Setzekorn's program, which uses a modified version of the tangent offset method.

The first user input is the number of stations in the section. This is limited to a maximum of fifty. Next the user inputs the following information on each of the stations being designed:

1. Station Number
2. Is the Curve a P.C. Station? If so, does the horizontal curve skew off to the right hand side or the left hand side?
3. Station elevation on the left side (28.1 feet to the left of the centerline).
4. Station elevation at the centerline.
5. Station elevation on the right side (28.1 feet to the right of the centerline).

The second input parameter, dealing with horizontal curve configuration, is input to allow the program to proportion superelevation along the curve transition length. This will allow the road cross sections to be gradually modified from a crowned road section at the P.C. and P.T. points to full superelevation sections within the horizontal curve.

Overall, it is important for the user to ensure all P.C. and P.T. points, as well as all hundred-foot stations are inputted. Failure to input all station data will result in errant vertical alignment output.

After all station input has been received, the program outputs this data in tabular form as follows:

1. Station Number
2. Left Side Elevation
3. Centerline Elevation
4. Right Side Elevation
5. Slope

The slope output at any one station is the straight line percent slope between that station and the previous station. Negative slopes indicate a decrease in existing centerline grade. It is important that the user review these slopes before the input of final grade. Proper review of this data will minimize cut and fill requirements on the final road cross sections.

Next the program requests from the user the final grade percent slope (G_i) between each of the stations. After input for all stations has been received the program requests the final grade elevation of the first station. From this data the program internally generates a straight-line final grade profile for all stations by the following formula:

$$\text{Sta Elev}_{i+1} = ((\text{Sta Dist}_{i+1} - \text{Sta Dist}_i) \times G_i) + \text{Sta Elev}_i \quad 6.24$$

The program then determines changes in final grade between consecutive stations and notifies the user of the need for a vertical curve at the inflection points (V). The user must then review the final grade input and select a length of vertical curve (L). Vertical curve alignment data is then generated using the following formulas:

$$y = r/2 x^2 + g_1 x + \text{Elev BVC} \quad 6.25$$

$$r = g_2 - g_1/L \quad 6.26$$

where: y = Station elevation at station $i + 1$
 x = Distance between station $i + 1$
and station i (in hundred feet).
 g_1 = Percent grade between the Beginning
station of the Vertical Curve (BVC).
and the grade inflection point (V).
 g_2 = Percent grade between the Ending
station of the Vertical Curve (EVC)
and the grade inflection point (V).
 L = Length of the vertical curve in
hundred feet.

A sample of such a vertical curve and the parabolic elevation calculations is shown by Figure 6.27. These final grade station elevations are then stored for use on the cross section design stage.

The Cross section design stage computes a standard cross section for each of the stations. This standard cross section is depicted in Figure 6.28. The cross sectional distances of these station cross sections are permanent. However, the relative slope and elevation of the cross section reference points change as a road section passes through a horizontal curve.

On tangent sections, the program computes a crowned cross section through each of the reference points. These are defined as follows:

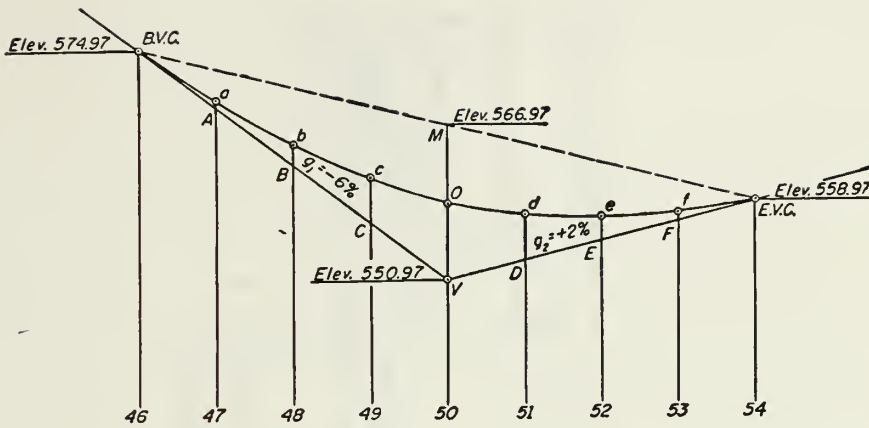
$$\text{Pvmt Edge Elev} = \text{Centerline Elev} - (11.5' \times 0.042) \quad 6.28$$

$$\text{Shoulder Elev} = \text{Pvmt Edge Elev} - (4.0' \times 0.063) \quad 6.29$$

$$\text{Ditch Elev} = \text{Shoulder Elev} - (4.5' \times 0.333) \quad 6.30$$

$$\text{Backside Elev} = \text{Ditch Elev} + (8.1' \times 0.275) \quad 6.31$$

As crowned road sections are symmetrical, these equations apply



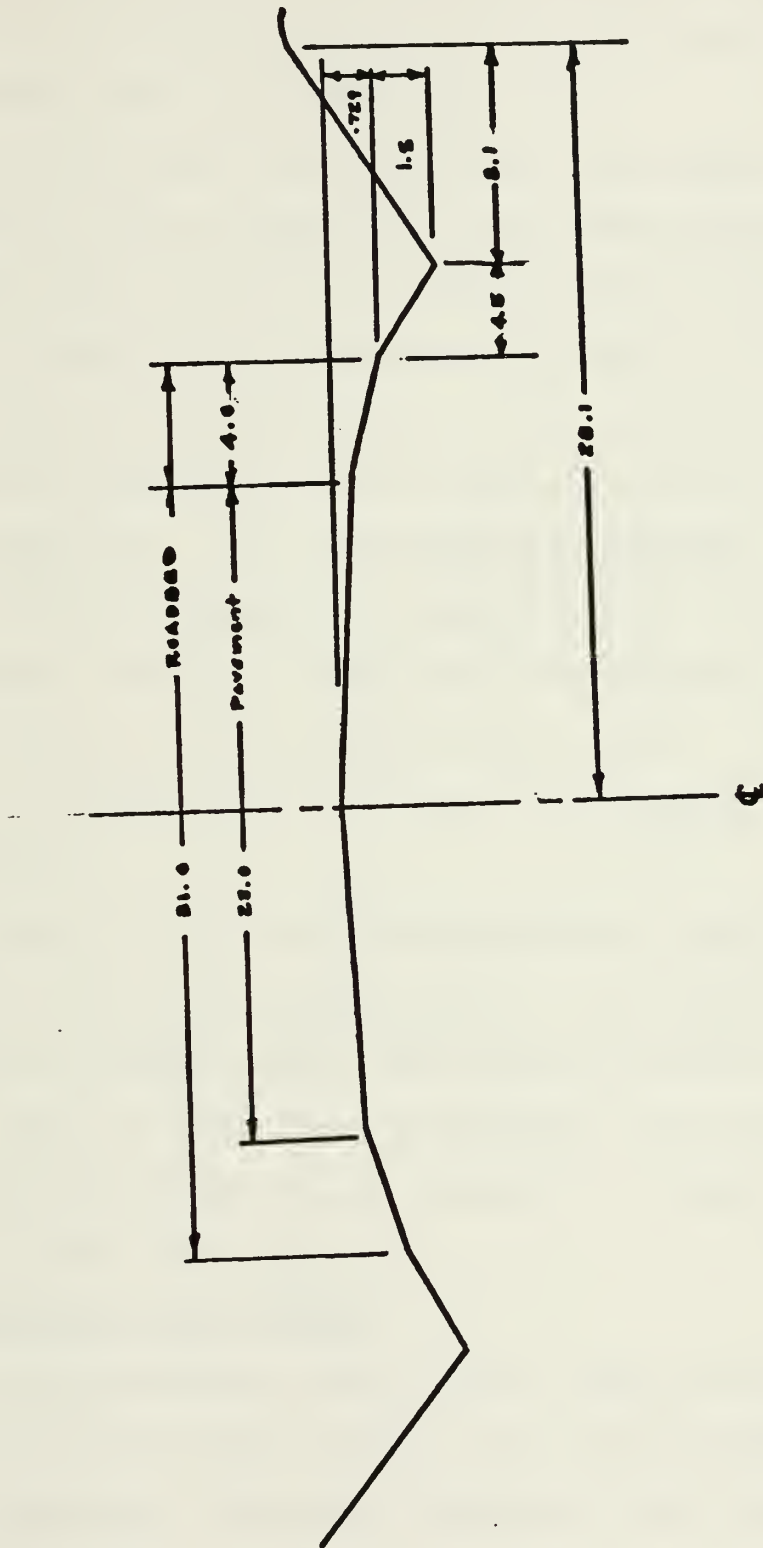
Typical Vertical Curve

Station	x	x^2	$\frac{r}{2}x^2$	g_1x	Elev. B.V.C.	Elev. Curve
B.V.C. = 15	0	0	0	0	882.35	882.35
16	1	1	- 0.33	+ 1.25	882.35	883.27
17	2	4	- 1.33	+ 2.50	882.35	883.52
18	3	9	- 3.00	+ 3.75	882.35	883.10
19	4	16	- 5.33	+ 5.00	882.35	882.02
20	5	25	- 8.33	+ 6.25	882.35	880.27
E.V.C. = 21	6	36	- 12.00	+ 7.50	882.35	877.85

Parabolic Elevation Calculations

Vertical Curve

Figure 6.27³⁰



N.T.S.

Standard Cross Section

Figure 6.28¹

to both sides of the road in cut. If the road is in a fill section, the standard cross section stops at the shoulder and is sloped down to the fill toe.

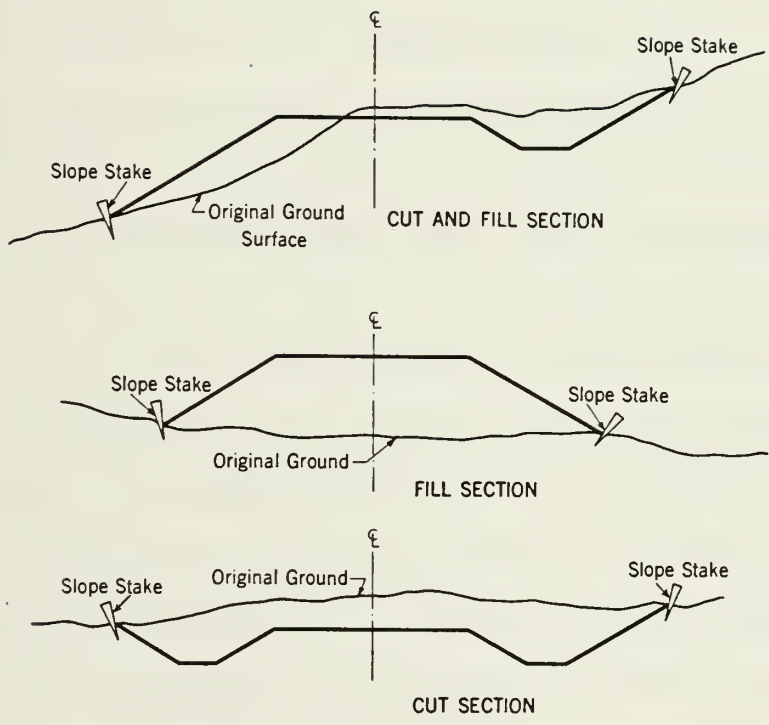
At horizontal curve sections, the program applies the super-elevation ratios developed in the second design stage. The program first calculates a superelevation transition ratio. This is defined as a ratio between the P.C. or P.T. of the curve and the curve midpoint. In this way superelevation is gradually applied until full superelevation is achieved at the curve midpoint. Then superelevation gradually tapers off as the distance from the curve midpoint to the point of tangency increases. Finally, at the point of tangency, the cross section returns to the standard cross section. If no superelevation is required in the curve, the standard cross section is applied.

Output for the cross section design stage is produced in tabular form. The program outputs the backside, ditchline, shoulder, pavement edge, and centerline elevations at each station. Due to paper size limitations, the program is formatted to output this data in two sections: as elevations left and right of the centerline.

Intersection Point Design

The intersection point design stage was developed to determine the catch points between the existing grade and the road-bed sideslopes. These are designed as the slope stake points on Figure 6.32.

User input consists of a slope grade for every station



Intersection Points

Figure 6.32²⁹

on both the right and left side of the standard cross section. As noted by Figure 6.32 this slope grade starts at the shoulder edge and slopes down to the slope toe on fill sections, and starts at the backside and slopes upward on cut section. The user inputs an integer which indicates to the program a slope ratio. For instance, a slope grade of four indicates the slope is one foot vertical to four foot horizontal (1:4). From this ratio, the program is able to calculate the intersection point between the design section and the existing grade.

The program determines each slope intersection point by an approximation method. First the program compares the existing and design shoulder edge elevations to determine whether the section is in cut or fill. The program then calculates the slope of the existing ground by interpolating between the existing ground centerline and left/right side elevations. If the design slope grade is found to be parallel with the existing slope grade, an error message is given and the user is instructed to select another design slope grade. If the slopes are not parallel, the program increments the slopes out, one horizontal foot at a time, until the elevations of the slopes either meet or cross one another. An error of less than one square foot per cross section is expected using this method.

Program output consists of the station number, grade elevation, and horizontal distance from the centerline; for both the right and left hand side slopes at each station. The data is presented in tabular form.

The cross sectional area design stage was developed to determine the cross sectional area of the cut or fill at each station. The design procedure utilized is known as the criss-cross method. This method treats each station cross section as a traverse. Each point of inflection on that traverse is given X and Y coordinates in relation to a reference point on the cross section. For the purpose of this program the left hand side intersection point was designated as the origin point. The program computes the cross-sectional area by use of the following equation:

$$\text{Double Area} = \sum_{i=1}^n Y_i (X_{i+1} - X_{i-1}) \quad 6.33$$

where i equals the number of inflection points on the traverse.

An example of this calculation is given by Figure 6.34.

The calculation of the volume and cumulative volume at each station is also accomplished in the cross sectional area subroutine. The volume calculation utilizes the average end area method. According to this formula, the volume, in cubic yards, between two cross sections having areas A_0 and A_1 is:

$$V = \frac{1}{2}(A_0 + A_1)L/27 \quad 6.35$$

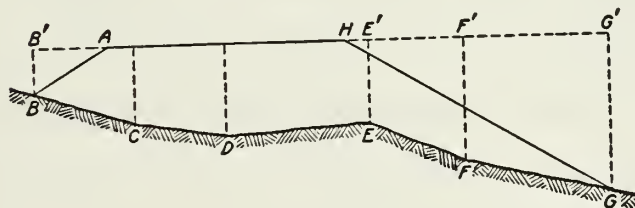
where: L is the distance between stations

First the program designates each cross section area as negative or positive, depending upon whether the section is a cut or fill respectively. The program then calculates the volume between each station using the average end area formula. Finally, the program sums all the volumes on a station by station basis, to get the cumulative volume between each station. The cumula-

A	B	C	D	E	F	G	H
<u>F 0.0</u>	<u>F 5.2</u>	<u>F 6.8</u>	<u>F 7.2</u>	<u>F 6.1</u>	<u>F 7.4</u>	<u>F 9.6</u>	<u>F 0.0</u>
- 12.0	- 19.8	- 10.0	0	+ 15.0	+ 20.0	+ 26.4	+ 12.0
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)

$Y_n (X_{n+1} - X_{n-1})$	= Double area
0.0 (- 19.8 - 12.0)	= 0.0
5.2 (- 10.0 + 12.0)	= + 10.4
6.8 (0.0 + 19.8)	= + 134.6
7.2 (+ 15.0 + 10.0)	= + 180.0
6.1 (+ 20.0 - 0.0)	= + 122.0
7.4 (+ 26.4 - 15.0)	= + 84.4
9.6 (+ 12.0 - 20.0)	= - 76.8
0.0 (- 12.0 - 26.4)	= + 0.0
	<u>2)454.6</u>
Area =	227.3 sq ft

Typical Criss-Cross Method
Cross Sectional Area Calculation



Cross Section in Question

Cross Sectional Area

Figure 6.34 ³⁰

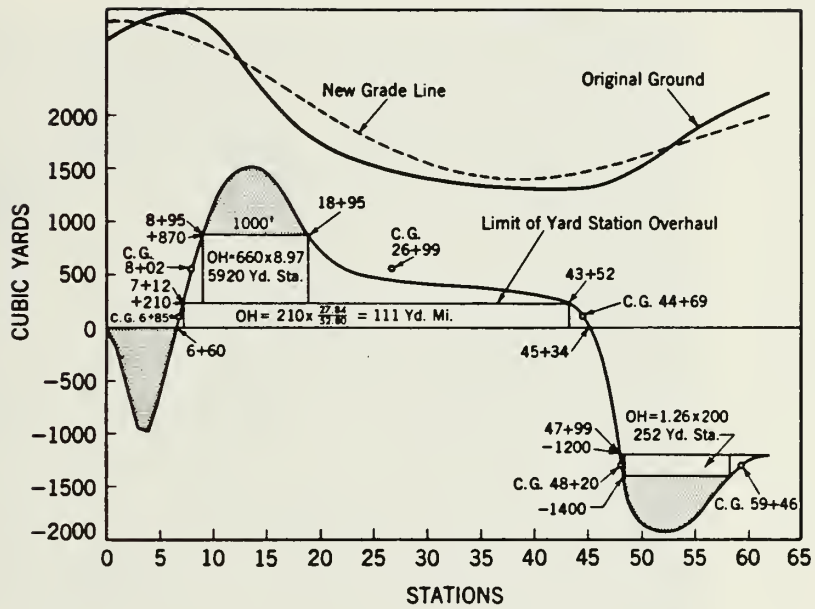
tive volumes will be used later in the program to plot a mass balance diagram.

The output from the cross section design stage consists of the station number and cross sectional area (in square feet) at each station, and the volume and cumulative volume (in cubic yards) between each station. The data is presented in tabular form.

Mass Balance Development

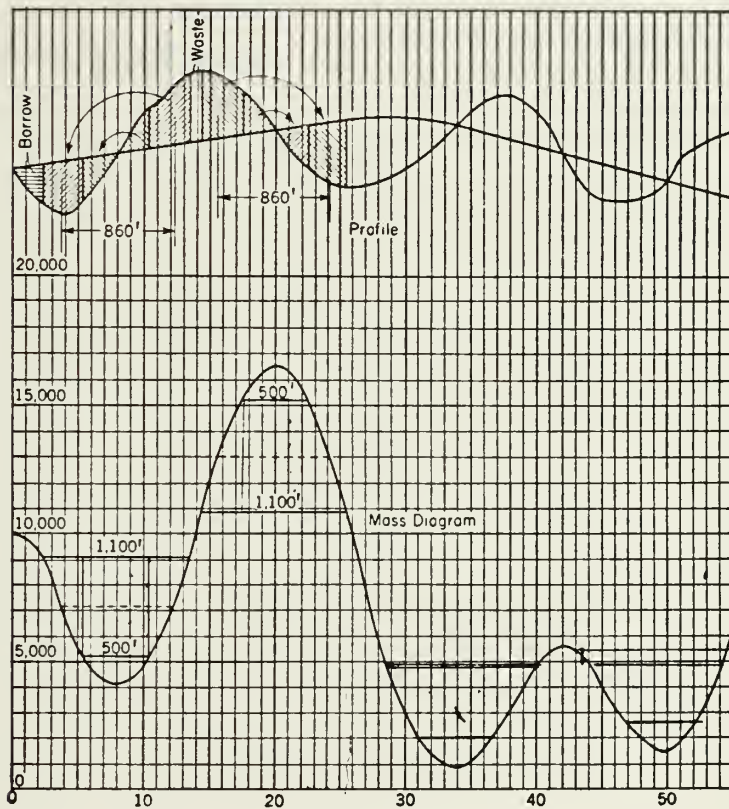
The final design stage, mass balance, determines the amount and location of the maximum and minimum cumulative volumes. Further it plots the cumulative volume versus the station numbers on the Versatec plotter. This plot is known as a mass diagram. An example of such a plot is given by Figure 6.36.

The mass diagram is utilized by construction managers to determine the amount of borrow and waste generated by the earthwork design. As shown by Figure 6.36, a typical road design contains both cut and fill sections. Often it is possible to use a portion of the cut section earthwork volume on the fill sections. This is shown on Figure 6.37. This eliminates the need to borrow more fill material than is necessary on a job. Further the construction manager can determine what amount of excess cut, if any, will have to be wasted (carried off the job). In civilian practice, the mass diagram is sometimes even more useful than a material distribution device. Some construction contracts limit the distance the contractor is expected to haul material taken from excavations to form embankments. This distance is known as the limit of free haul. Any distance



Mass Diagram

Figure 6.36³⁰



Mass Diagram Showing Borrow
And Waste Volumes

Figure 6.37²⁹

in excess of this is known as overhaul, and is associated with a recompensable expense.

In this program, the maximum and minimum cumulative volumes are determined by means of a sort routine. This output data is presented in written form on the printer. This is the last data output by the line printer. The Versatec plotter plots the mass diagram output. Cross marks indicate the amount of cumulative volume accrued at each station. After output of this data, the program ends.

CHAPTER VII

CONCLUSIONS

The need for computer aided designs in the field of combat engineering is both present and pervasive in the Armed Forces. Current research in this area quite correctly centers around the use of small, mobile computers; even if the state of the art in the computer manufacturing world has not advanced sufficiently to produce a model which can endure the rugged field conditions. Many of the CAD's developed by the Navy and at the Georgia Institute of Technology have dealt with the construction problems facing the military engineer in an Amphibious Objective Area. Program FIRD can now be added to that list.

Though program FIRD was prepared on the Cyber 70 system at the Georgia Institute of Technology, an effort was made to develop the program as a series of subroutines. Each of these subroutines are fairly inclusive, with both the input and output of each subroutine developing a design stage in its entirety. No single subroutine is excessively long, or beyond the storage capacities of most microprocessors. Most importantly, the design methods utilized by the program are not complex. Only simple curves are designed in the Horizontal Alignment subroutine, the CBR method is the most common procedure in pavement design, and the criss-cross method of determining the cross-sectional area of a traverse is the most flexible method available. These, among other factors,

lend themselves to the concept of translating program FIRD to a microprocessor for the benefit of a combat engineer. The only real problems facing such an endeavor are associated with the translation of the present Fortran IV program coding to the actual microprocessor language. This is especially noteworthy in the Mass Balance subroutine, where the Cyber system requires highly specialized "calls" to activate the Versatec plotter.

The overall program, though crude, is both servicable and safe. A listing of the program is given by Appendix E. There are some limitations to this program. The most important of these pertain to the friendliness of the system. There are many input parameters to be entered in a road design (especially in the Vertical Alignment design stage). It would be advisable to redesign some of the input formatting such that errant entries can be easily corrected without jeopardizing the entire program run. In addition, if the program is to be maintained in its entirety, a "menu", leading the user to each of the design stages should be added. Finally, some provision for the input of data by tape should be included. It was felt that these items might better be addressed by the needs of the user.

Some of the individual subroutines have built-in design limitations. The superelevation subroutine pro-rates the superelevation along the entire length of the horizontal curve. The only time full superelevation is reached is at the midpoint of the curve. It may be necessary to pro-rate the superelevation upon a shorter transition length, in the case of a long hori-

zontal curve. The determination of the intersection points between the existing ground and design road section uses an approximation method. This can lead to an error of up to one square foot per road section. The implementation of a linear regression method would eliminate this error. The cross-sectional area calculation does not provide for the depth of the pavement. This leads to slightly higher areas of fill and slightly lower areas of cut than truly exists. This was overlooked in the program design stage of the development of this research.

Finally, the quality of the mass balance curve from the Versatec plotter is somewhat below average. Sample runs produced graphs somewhat off in the plotting of the station locations. It is thought that this is at least partly due to hardware problems in the plotter itself. The message below seems to confirm this assertion.

```
NEWS FOLLOWS...  
LAST NEWS RUN ON      81/11/03. 10.00.53.
```

```
81/11/03. 19.09.40.
```

```
  DUE TO A HARDWARE PROBLEM, THE QUALITY OF  
  VERSATEC PLOTS WILL BE BELOW AVERAGE. OCS  
  HOPES TO CORRECT THIS PROBLEM BY NOON WEDNESDAY.
```

```
81/11/03. 16.16.35.
```

```
  THE CALCOMP PLOTTER IS DOWN DUE TO HARDWARE  
  FAILURE. THE CALCOMP SHOULD BE UP ON 81/11/04  
  BY 1800.
```

Due to time limitations, this matter was not investigated more fully. A sample run indicating this problem, as well as demonstrating the capabilities of the entire program, is provided by Appendix F.

There are many ways program FIRD can be enhanced. Plots of the vertical alignment and station cross sections may be required. It may be desirable (though certainly not easy) to output all the plots on a line printer. Even more practical to the combat engineer however, would be subroutines which determine operator and equipment requirements, as well as durations, based upon the cut and fill volumes. Perhaps some interface between program FIRD and the MAFESS program of Lt. D. Pruett might eventually be developed.

REFERENCES

1. Setzekorn, Robert R., Computer Aided Road Design Procedures for Microprocessors, unpublished report, Georgia Institute of Technology, Atlanta, Georgia, March, 1981.
2. "ADPE Acquisition of a Mini Computer for 20th Engineer Brigade (Combat) (Airborne Corps)," 20th Engineer Brigade Letter AFZA-EN-B-O dated 14 July 1980.
3. "Standard ADPE Justification for 35th Signal Brigade Tactical Computer System," Appendix J to AR 18-1 dated 22 March 1980.
4. Hambrock, Daniel L., Computer-Aided Design of Heliports and Expeditionary Airfields in the Amphibious Operations Area, unpublished report, Georgia Institute of Technology, Atlanta, Georgia, June, 1981.
5. TRI-TAC, Joint Tactical Communications Office Pamphlet, September, 1979.
6. Department of the Navy, Seabee Planner's and Estimator's Handbook, NAVDOCKS P-405, U.S. Government Printing Office, Washington, D.C., 1974.
7. PDP-11, Microcomputer Members of the PDP-11 Family, Digital Equipment Corporation Brochure dated 1980.
8. Civil Engineer Support Office, Construction Management CM-4 Program, Program Documentation, Naval Construction Battalion Center, Port Hueneme, Ca., August, 1980.
9. Civil Engineer Support Office, Material Liaison Office Program, System Documentation, Naval Construction Battalion Center, Port Hueneme, Ca., November, 1980.
10. Civil Engineer Support Office, People Minicomputer Program, Naval Construction Battalion Center, Port Hueneme, Ca., December, 1980.
11. Civil Engineer Support Office, Medical Minicomputer Program, Naval Construction Battalion Center, Port Hueneme, Ca., December, 1980.
12. Civil Engineer Support Office, Tool Minicomputer Program, Naval Construction Battalion Center, Port Hueneme, Ca., December, 1980.

13. United States Marine Corps, I Marine Amphibious Force Order P5230.8, I Marine Amphibious Force FMF, Camp Pendleton, Ca., dated December, 1980.
14. Operational and Organizational Concept for Division Level Data Entry Device (DLDED), A Study by U.S. Army Signal Center and Ft. Gordon, Ft. Gordon, Georgia, 19 December, 1980.
15. U.S. Marine Corps, Engineer Operations, FMFM 4-4, U.S. Government Printing Office, Washington, D.C., March, 1979.
16. Springston, P.S., Earthwork Construction in Support of A Marine Amphibious Force - A Case Study, Technical Note, TN no. N-1514, Civil Engineering Laboratory, Naval Construction Battalion Center, Port Hueneme, Ca., January, 1978.
17. GSA Task order #92-44, "U.S. Marine Corps MAF Engineer Development Engineering Planning System Functional Description," Civil Engineering Laboratory, Port Huene-me, Ca., May, 1980.
18. Civil Engineering Laboratory, U.S. Marine Corps Engineers Amphibious Objective Area Land Management System (AOALMS): Users Manual, Naval Construction Battalion Center, Port Hueneme, Ca., undated.
19. Anderson, Joseph, "A Prototype Decision Support System for the Location of Military Water Points," M.S. Thesis, Industrial and Systems Engineering Reports Series, Georgia Institute of Technology, Atlanta, Georgia, June, 1980.
20. Durham, Dwight S., Use of Computerized Network Analysis Systems on U.S. Army Corps of Engineers' Construction Projects, Unpublished Report, Georgia Institute of Technology, Atlanta, Georgia, December, 1980.
21. Halpin, D.W., A Computerized Construction Management Gaming Environment, SCEGIT-75-114, Georgia Institute of Technology, December, 1975.
22. Halpin, D. W., Computer Aided Design for MAF Engineers, SCEGIT-80-193, Georgia Institute of Technology, Atlanta, Georgia, December, 1980.
23. Department of the Army, TM5-330, Planning and Design of Roads, Airbases, and Heliports in the Theater of Operations, U.S. Government Printing Office, Washington, D.C., 1974.

24. Pruett, David D., Equipment Estimating and Scheduling for the Marine Amphibious Force, unpublished report, Georgia Institute of Technology, Atlanta, Georgia, June, 1981.
25. Bennett, F. Lawrence, Critical Path Resource Scheduling Algorithm, Journal of the Construction Division, ASCE, Vol. 94, No. C02, Proc. Paper 6152, October, 1968.
26. The Asphalt Institute, Soils Manual for Design of Asphalt Pavement Structures, MS10, College Park, Maryland, 1969.
27. Department of the Army, TM5-824-2, Airfield Flexible Pavements Air Force, U.S. Government Printing Office, Washington, D.C., 1976.
28. Department of the Army, TM5-337, Paving and Surfacing Operations, U.S. Government Printing Office, Washington, D.C., 1977.
29. Paquette, Radnor J. and Wright, Paul H., Highway Engineering 4E, John Wiley and Sons, 1979
30. Bouchard, Harry, et al, Surveying, Sed., International Textbook Company, 1965.

APPENDIX A

Programs Developed by US Army Engineer School

1. BAILEY BRIDGE - This program computes the type of bridge and its installation time, given the gap width, bank elevations, soil conditions and required load class.
2. BRGCLASS - This program classifies wood and steel stringer spans up to 60 feet in length.
3. BRGDES - This program computes the size of wood or steel stringers for fixed span bridges.
4. CPM - This program analyzes critical path method networks.
5. CRUSH - This program computes the jaw and roll settings and screen dimensions for 75 TPH rockcrusher.
- * 6. DEMO - This program computes charges and installation times for specific explosive applications.
7. DITCH - This program computes ditch sizes given soil conditions, flow and slope.
8. EXTWIRE - This program computes the power distribution requirements from generator sources.
9. FLEXIBLE PAVEMENT - This program computes pavement and subbase design to include CBR, compaction, and thickness.
10. H CURVE - This program computes horizontal curve data.
11. MASABT - This program computes mass concrete bridge abutments.
- *12. MINFLD - This program computes the logistical and man-power requirements for breaching and clearing minefields.
- *13. OBSTACLE - This program computes materials, troop effort, and equipment requirements for specified obstacles in a barrier plan.
14. QUARRY - This program computes blast patterns for rock quarries.

*Program with non-construction application.

15. RIVFUN - This program computes floating bridge and raft requirements.
16. STLSTR - This program computes steel beam stringer sizes for fixed span bridges.
17. VERTCURV - This program computes vertical curve elevations.
18. WOODSTR - This program computes wooden stringer sizes for timber trestle bridges.

APPENDIX B

Programs Developed by 20th Engineer Brigade

- * 1. CHISQR - Computes the CHI square statistic for non-parametric testing of independence of two variables.
2. CONNAN - Tracks all construction projects in the brigade. Interacts with EQCOST and PROJ COST.
3. CROSSEC - Computes the cross-sectional area of a polygon by utilizing the distance from centerline and relative elevation of each point in the polygon.
4. CURVE - Provides a computer profile plot of topographic elevations and computes slopes between stations.
5. EARTH1 - Provides a graphical plot of the topographic lines and design grades with all slopes and elevations. Additionally, it provides a mass diagram.
6. EARTH 25 - Controls a system of programs to obtain topographic and grade profiles as well as earthwork estimates and mass diagrams using random horizontal distances.
7. EQCOST - Stores data on rental and depreciation costs.
8. EQSTAT - Computes daily equipment status of selected items of equipment. The report reflects the number on hand, number committed, number available, the deadline rate, and commitment rate based on number operational.
- * 9. GSEARCH - Searches an obstacle list for obstacles within a specific area and produces a listing containing logistical information.
- *10. LOGSHIP - Provides weight and dimensions on table of equipment items for rail, sea, and air deployment.
- *11. LOGSTAT - Provides listings of table of equipment by line number sequence. Listings can be grouped by commodity code and the total items or a combined list of several units can be generated.

*Program with non-construction application.

12. LOGSTAT3 - Produces logistical status reports in the form required by Third Corps.
13. MANPOWER ANALYSIS - Records, categorizes and plots trends in daily manpower utilization.
- *14. MINEFLD2 - Assists in the planning and operational aspects of standard pattern minefield installation.
- *15. OBS - Computes logistical and manpower requirements for a Brigade Obstacle Plan.
16. PROJCOST - Records manhours and equipment hours to compute funded and unfunded project costs.
- *17. RAFTOPN2 - Produces a complete time table given the rafting assets and equipment to be crossed.
- *18. ROSTER - Updates, sorts and prints the Brigade's officer roster.
19. RUNWAY - Provides a graphical representation of landing and clear zones requirements for C-130 aircraft.
- *20. STEVRS/STDEVGE - Calculates the mean and standard deviation of random sample and grouped frequency data respectively.
- *21. SQTP - Prints an individual letter to each soldier to inform him of the SQT test which he is to take and prints a roster for each unit to inform the commander of all personnel in his unit that will be tested.

*Program with non-construction application.

APPENDIX C

USER'S MANUAL FOR PROGRAM FIRD

USER'S MANUAL

INTRODUCTION

This user's manual contains step-by-step procedures to operate program FIRD. As FIRD is an interactive program, the computer provides questions which you, the user, must answer. Many of these questions are technical decisions, usually handled by a highway engineer. Therefore the user may find it helpful to read Chapter VI - PROGRAM DESIGN, which delineates the engineering basis for these decisions, before inputting.

Program FIRD is currently loaded on the CYBER-70 system at the Georgia Institute of Technology. All procedures set forth in this manual (except log-on procedures) refer to the CYBER system. All the procedures in this manual have been numbered. In general, the steps indicating correct input are consecutively numbered (i.e., 1, 2, 3, etc.). Steps indicating errant input, design comments, or program output are numbered and alphabetized (i.e., 1A, 1B, 1C, etc.).

Before using the program, read all of the instructions carefully and assemble your input data. Good luck.

FIRD PROGRAM PROCEDURES

STEP 1

After logging on the computer, address program FIRD with the below job control commands. If a program listing is required interchange L = Ø in the FTN command with R = Ø.

```
OLD,FIRD
/ATTACH,CALCOMP/UN=LIBRARY
/LIBRARY(CALCOMP)
LIBRARY(CALCOMP)
/FTN,I=FIRD,L=Ø,TS
      16.308 CP SECONDS COMPILATION TIME
/LGO.
```

STEP 2

The program will initialize with some comments and proceed to the HORIZONTAL ALIGNMENT design stage. The first input data is the number of curves to be designed in the road section. A maximum of twenty curves can be designed at one time. This is input as an integer in I2 format (i.e., 1 for one curve). If no curves are to be designed, type in Ø and the program will move on to Step 5 (Soils Design).

THIS PROGRAM IS A SERIES OF SUBROUTINES DESIGNED TO ASSIST THE ENGINEER IN A MILITARY ROAD DESIGN. THE OPERATION OF THE PROGRAM IS INTERACTIVE, AND REQUIRES THE USE OF THE USERS MANUAL.

HORIZONTAL ALIGNMENT

ENTER NO. OF CURVES THIS SECTION

? 1

STEP 3

The program will request input on the tangent intersection point of the curve, the angle of intersection, and the chord length.

All three of these items are input as real numbers in F7.3 format (i.e., 133.333). From this data, the program calculates horizontal curve alignment and provides output in tabular form. From left to right this output reads as follows: curve midpoint station number, curve radius, length of curve, degree of curve, tangent length, beginning of curve (P.C.) station number, and end of curve (P.T.) station number.

ENTER STA NO OF TANGENT INTERSECTION OF CURVE 1

? 133.0

ENTER ANGLE OF INTERSECTION OF CURVE IN DEGREES

? 26.0

ENTER CHORD LENGTH IN FEET

? 600.0

STA NO.	RAD	L OF C	D OF C	TAN LEN	PC STA	PT STA
132.947	1333.62	605.181	9.46705	307.891	129.921	139.052

STEP 3A

As a safety feature, the program automatically checks on the degree of curvature (D). If the value of the degree of curvature exceeds 10.0, the program returns the user to Step 3 for curve redesign. This may be accomplished by decreasing the angle of intersection or increasing the chord length.

WARNING D TOO HIGH REESTABLISH PC, PT, ANDC

ENTER STA NO OF TANGENT INTERSECTION OF CURVE 1
? 133.0

ENTER ANGLE OF INTERSECTION OF CURVE IN DEGREES
? 26.0

ENTER CHORD LENGTH IN FEET
? 540.0

WARNING D TOO HIGH REESTABLISH PC, PT, ANDC

ENTER STA NO OF TANGENT INTERSECTION OF CURVE 1
? 133.0

ENTER ANGLE OF INTERSECTION OF CURVE IN DEGREES
? 26.0

ENTER CHORD LENGTH IN FEET
? 570.0

STEP 4

The program proceeds on to the SUPERELEVATION design stage. User input to this stage consists of the design velocity of the curve. The user has the option of 30, 40, 50 or 60 miles per hour as this design velocity. Data is inputted as an integer in I2 format (i.e., 60). From this data, the program calculates Superelevation data and provides output in tabular form. From left to right this output reads as follows: curve midpoint station number, curve radius, length of curve, rate of superelevation, tangent length, beginning of curve (P.C.) station number, and end of curve (P.T.) station number.

SUPERELEVATION SUPERL

WHAT IS DESIGN VELOCITY OF CURVE 1
? 60

STA NO	RAD	L OF C	SUPERL	TAN LEN	PC STA	PT STA
132.947	1333.62	605.181	.04996069	307.891	129.921	139.052

STEP 4A

If the velocity input is not one of the optional velocities, 30, 40, 50 or 60 miles per hour, the program will reject the data and return the user to Step 4.

WHAT IS DESIGN VELOCITY OF CURVE 1
? 80

INVALID VELOCITY-TRY AGAIN

WHAT IS DESIGN VELOCITY OF CURVE 1
? 40

STEP 4B

If the velocity input results in a calculated superelevation rate of between 0.08 foot per foot and 0.10 foot per foot, the program will ask if snow is prevalent in the area. If snow is prevalent (YES), the program returns the user to Step 3 for realignment of the horizontal curve. If snow is not prevalent in the area (NO), the program accepts the calculated superelevation rate and continues on to the SOILS design stage.

WHAT IS THE DESIGN VELOCITY OF CURVE 1
? 60.0

WARNING IS SNOW PREVALENT?
ENTER 1 IF YES AND 2 IF NO.
? 1

ENTER STA NO OF TANGENT INTERSECTION OF CURVE 1
? 133.0

STEP 4C

If the velocity input results in a calculated superelevation rate greater than 0.10 foot per foot, the program will output a warning and return the user to Step 3 for realignment of the horizontal curve.

WHAT IS DESIGN VELOCITY OF CURVE 1
? 60

WARNING-RADIUS MUST INCREASE TO DECREASE SUPERL OF CURVE 1
WARNING D TOO HIGH REESTABLISH PC, PT, ANDC

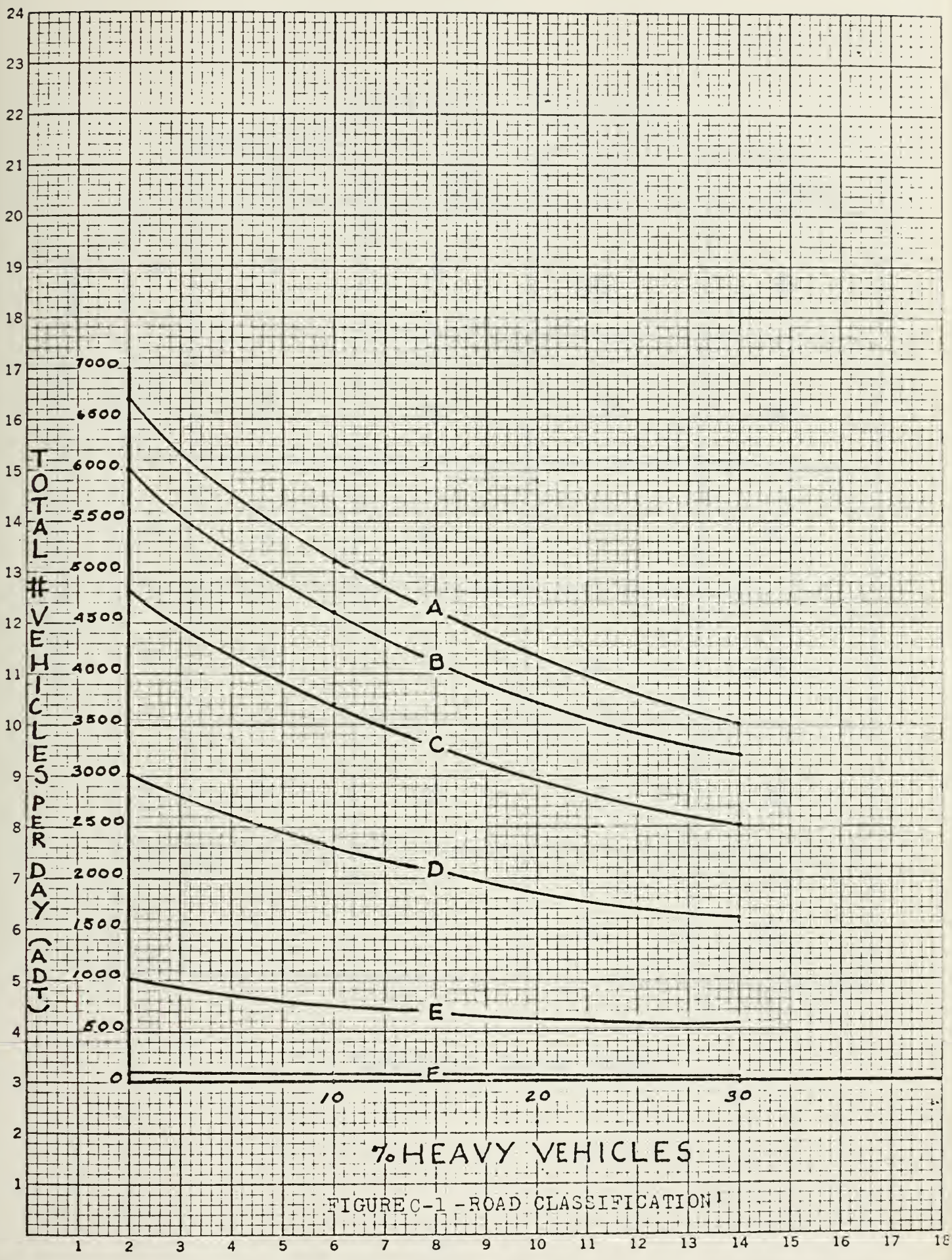
ENTER STA NO OF TANGENT INTERSECTION OF CURVE 1
? 130.0

ENTER ANGLE OF INTERSECTION OF CURVE IN DEGREES
? 50.0

ENTER CHORD LENGTH IN FEET
? 600.0

STEP 5

The program proceeds on to the SOILS design stage. To utilize this subroutine, the user must first estimate the Average Daily Traffic (ADT) and percentage of heavy vehicles expected on the roadway. By entering the graph of Figure C-1 with the ADT and percentage of heavy vehicles, the user can determine a suitable road classification. By entering the graph of Figure C-2 with the percentage of heavy vehicles and the road classification, the user can select the proper design index. These parameters are more fully explained in Chapter VI, PROGRAM DESIGN. Both the road classification and design index will be used as inputs to the program later on.



7% HEAVY VEHICLES

FIGURE C-1 - ROAD CLASSIFICATION

% HEAVY VEHICLES

ROAD CLASSIFICATION

	0	10	20	30
A	3	4	5	6
B	3	4	5	6
C	3	4	4	6
D	2	3	4	5
E	1	2	3	4
F	1	1	2	3

FIGURE C-2 - DESIGN INDEX ¹

STEP 6

Input the design index data of Step 5 into the program. This is input as an integer in I1 format (i.e., 6).

WHAT DESIGN INDEX IS TO BE USED?
? 6

STEP 7

Input the in-place density and California Bearing Ratio (CBR) of the existing subgrade into the program. The subgrade in-place density input data should be estimated as a percentage of maximum dry density utilizing a modified Proctor test (CE55). The subgrade CBR data should be estimated as if the subgrade were compacted from 90 to 95% of the maximum modified proctor density. Both data items must be input as real numbers (i.e., 70.0) as the program input format is F 4.1.

WHAT IS THE IN-PLACE (%CE55) OF THE SUBGRADE?
? 70.0
WHAT IS THE CBR OF THE SUBGRADE AT 90% TO 95% CE55?
? 5.0

STEP 7A

If the input in-place density of the subgrade is less than 20% or more than 100%, the program will output an error message and return the user to Step 7.

WHAT IS THE IN-PLACE (%CE55) OF THE SUBGRADE?
? 19.0

INPLACE DENSITY MUST BE GIVEN AS %CE55. TRY AGAIN.
WHAT IS THE IN-PLACE (%CE55) OF THE SUBGRADE?
? 20.0

STEP 7B

If the input CBR of the subgrade is less than 2 or more than 50, the program will output an error message and return the user to Step 7.

WHAT IS THE CBR OF THE SUBGRADE AT 90% TO 95% CE55?
? 120.0

VALUES FOR CBR SHOULD RANGE FROM ABOUT 2 TO 50. TRY AGAIN.
WHAT IS THE CBR OF THE SUBGRADE AT 90% TO 95% CE55?
? 15.0

STEP 8

Input the number of base, subbase, and select material layers to be utilized in the pavement design. This data is input as an integer in I1 format (i.e., 3).

HOW MANY TYPES OF BASE, SUBBASE, AND SELECT MATERIALS
DO YOU WISH TO CONSIDER? (DO NOT INCLUDE SUBGRADE OR PAVT)
? 3

STEP 8A

If the number of layers is greater than 7 or input as 0, the program will output a warning and return the user to Step 8.

HOW MANY TYPES OF BASE, SUBBASE, AND SELECT MATERIALS
DO YOU WISH TO CONSIDER?(DO NOT INCLUDE SUBGRADE OR PAVT)
? 8

THIS PROGRAM WILL HANDLE UP TO 7 DIFFERENT SOILS.
THIS SHOULD BE MORE THAN SUFFICIENT.

HOW MANY TYPES OF BASE, SUBBASE, AND SELECT MATERIALS
DO YOU WISH TO CONSIDER?(DO NOT INCLUDE SUBGRADE OR PAVT)
? 3

STEP 9

Input the CBR values for the base course, the subbase, and each
of the select materials chosen. These CBRs should be input
as real numbers (i.e., 82.0) as the program input format is F5.1.

WHAT IS THE CBR VALUE OF THE BASE COURSE?
? 82.0
WHAT IS THE NEXT HIGHEST CBR?
? 56.0
WHAT IS THE NEXT HIGHEST CBR?
? 43.0

STEP 9A

If the input CBR values are less than 2 or greater than 100,
the program will output an error message and return the user to
Step 9.

WHAT IS THE CBR VALUE OF THE BASE COURSE?
? 99.0
WHAT IS THE NEXT HIGHEST CBR?
? 101.0
VALUES FOR CBR SHOULD RANGE FROM ABOUT 2 TO 100.
TRY AGAIN.

STEP 9B

If the CBR values are not input in descending order, the program will output an error message and return the user to Step 9.

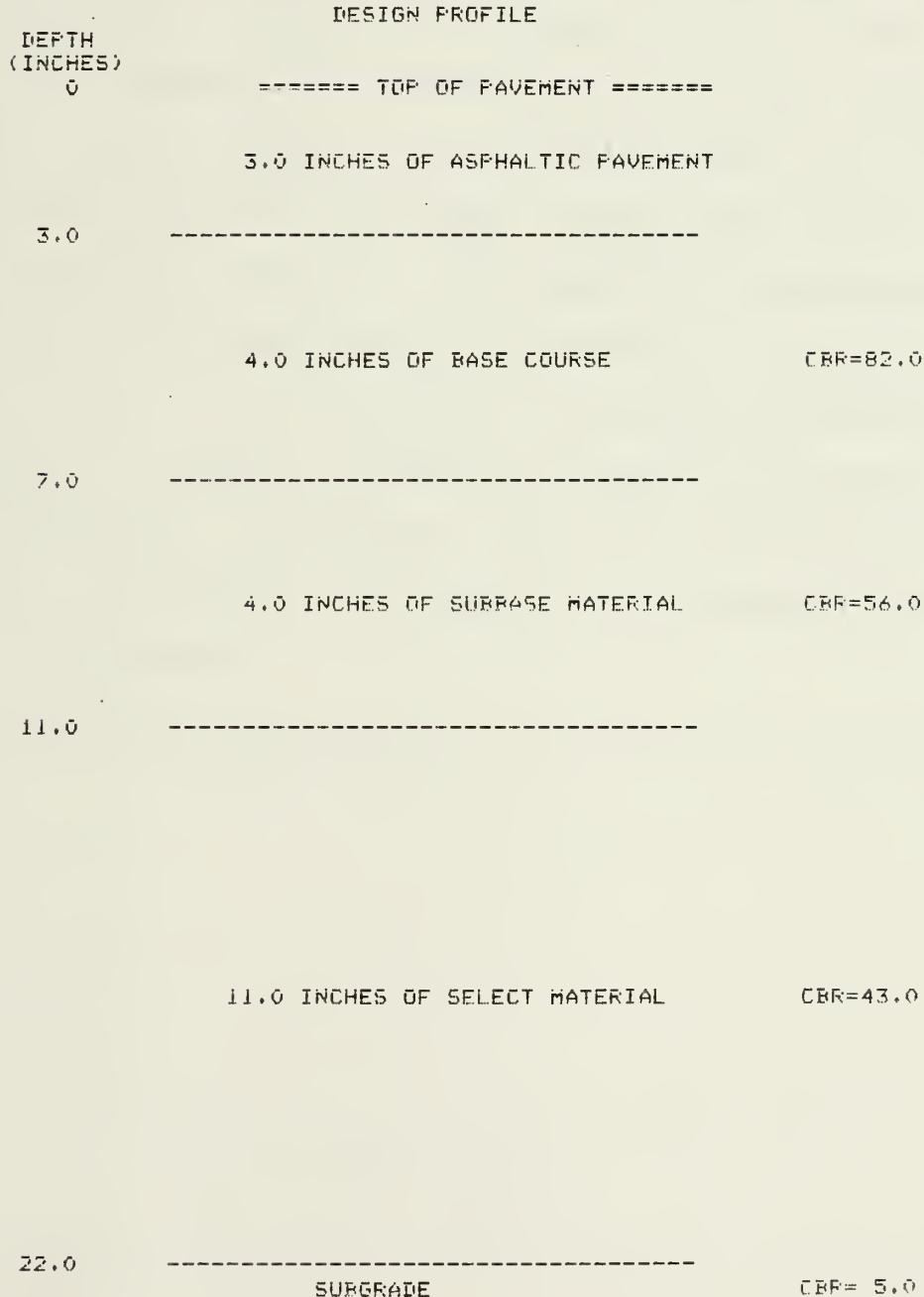
```
WHAT IS THE NEXT HIGHEST CBR?
? 90.0
WHAT IS THE NEXT HIGHEST CBR?
? 95.0
CBR VALUES WERE NOT GIVEN IN DESCENDING ORDER.
TRY AGAIN.
WHAT IS THE CBR VALUE OF THE BASE COURSE?
? 99.0
WHAT IS THE NEXT HIGHEST CBR?
? 95.0
WHAT IS THE NEXT HIGHEST CBR?
? 90.0
WHAT CLASSIFICATION IS TO BE USED?
A=1,B=2,C=3,D=4,E=5,F=6
? 3
```

STEP 10

The next data to be input is the road classification. This data is input as an integer in I1 format (i.e., 3). After this data has been input, the program outputs the minimum thickness requirements of the various layers of the pavement structure, including asphaltic pavement. The program also outputs a design cross section of a typical pavement section. This cross section is roughly scaled such that one inch of cross section material is represented by four computer lines.

WHAT CLASSIFICATION IS TO BE USED?
A=1,B=2,C=3,D=4,E=5,F=6
? 3

MIN THICKNESS RQD ABOVE SOIL WITH CBR= 5.0 IS 22.0
MIN THICKNESS RQD ABOVE SOIL WITH CBR=43.0 IS 4.0
MIN THICKNESS RQD ABOVE SOIL WITH CBR=56.0 IS 3.0
THE MIN RQD THICKNESS OF PAVT IS 3.0
MIN RQD THICKNESS OF BASE COURSE IS 4 INCHES



STEP 11

The last data input into the SOILS subroutine is a rating of the structural capability of the subgrade. The user indicates whether the soil is cohesive (silt or clay) with a Plastic Index (PI) greater than five or not. An index of this parameter is inputted as an integer in I1 format (3 or 4). After this data has been received, the program outputs the soil compaction requirements, and the design cross sections for both fill and cut sections. These are presented in the same, scaled cross section format as was discussed in Step 11. The compaction requirements schedule indicates the compaction as a percentage of the modified Proctor test (CE 55). The cut and fill cross sections indicate the quantity (in inches) and percent compaction for each soil layer being designed. In addition, the output provides special instructions, as necessary, with regard to the pavement surface treatment and subgrade.

IF THE SUBGRADE IS COHESIVE (PI>5), TYPE 3
IF THE SUBGRADE IS NON-COHESIVE, TYPE 4.
T 4

COMPACTION REQUIREMENTS (WRITTEN)

DEPTH .
(INCHES) ===== TOP OF PAVEMENT =====

100.0% CBR=82.0

7.0 -----

100.0% CBR=56.0

11.0 -----

100.0% CBR=43.0

22.0 -----

95.0% CBR= 5.0
TDF (TO DEPTH OF FILL)

FINAL DESIGN
FOR FILL SECTIONS

DEPTH
(INCHES) ===== TOP OF PAVEMENT =====

3.0 INCHES OF ASPHALTIC PAVEMENT

3.0 -----

4.0 INCHES, CBR= 82.0 AT 100% CE55

7.0 -----

4.0 INCHES, CBR= 56.0 AT 100% CE55

11.0 -----

11.0 INCHES, CBR= 43.0 AT 100% CE55

22.0 -----

TO DEPTH OF FILL, CBR= 5.0 AT 95.0% CE55

TDF -----

6 INCHES, SCARIFY AND COMPACT IN PLACE AT 75.0% CE55

CBR= 5.0 IN-PLACE DENSITY=70.0% CE55

DEPTH (INCHES) ===== TOP OF PAVEMENT =====

3.0 INCHES OF ASPHALTIC PAVEMENT

3.0 -----

4.0 INCHES, CBR= 82.0 AT 100% CESS

7.0 -----

4.0 INCHES, CBR= 56.0 AT 100% CESS

11.0 -----

11.0 INCHES, CBR= 43.0 AT 100% CESS

22.0 -----

SUBGRADE REMOVED AND REPLACED AT 100.0% CESS

28.0 -----

SUBGRADE REMOVED AND REPLACED AT 95.0% CESS

40.0 -----

SUBGRADE REMOVED AND REPLACED AT 90.0% CESS

60.0 -----

SCARIFY AND COMPACT IN-PLACE AT 85.0% CESS

66.0 -----
CBR= 5.0 IN PLACE DENSITY= 70.0% CESS

THIS COMPLETES THE SOILS DESIGN

STEP 12

The program initializes the VERTICAL ALIGNMENT design stage by outputting comments. These comments are designed to aid the user in laying out the road section. The first data to be input is the number of stations to be considered in the road section. This number should be input as an integer (i.e., 16) as I2 input format was used. A maximum of fifty stations can be designed at one session.

THIS SUBROUTINE WILL CALCULATE THE CENTERLINE
ELEVATION OF THE FINISHED GRADE. YOU MUST SUPPLY THE
ACTUAL ELEVATIONS OF THE LEFT SIDE DITCH BACKSLOPE,
CENTERLINE, AND THE RIGHT SIDE DITCH BACKSLOPE. THE
TERMINAL POINT FOR BACKSLOPES IS 20 FEET LEFT OR
RIGHT OF THE CENTERLINE. ENSURE THAT THE PC, PT, AND
MIDPOINTS OF ANY CURVES IN THIS SECTION ARE INCLUDED.

HOW MANY STATIONS IN THIS SECTION? MAX=50

? 16

STEP 13

Next the program requests data on each of the stations being designed. This data consists of the station number, the horizontal alignment, and the elevations of the existing road section. The station numbers should be input as real numbers (i.e., 127.0) as the program input format is F7.3. The station numbers and other station data, must be input in ascending order for the program to work properly. Further, it is important that the beginning station be input. As the horizontal alignment input

to the vertical alignment subroutine, the program asks the user whether the station number being input is a P.C. of a horizontal curve. If it is, the program requests input as to whether the horizontal curve bears to the right or left. This horizontal curve data is used as an index by the program to apply the super-elevation rates calculated in the SUPERELEVATION design stage to the road cross sections following the P.C. stations. As the super-elevation rates are stored in an array indexed by the horizontal curve number, if P.C. stations are not input in ascending order or if one of the P.C. stations is omitted, the wrong super-elevation rates will be applied to the cross sections. The horizontal curve data is input as an integer (0, 1 or 2) as the program input format is I2. The elevations of the existing road cross section are input as three data points. These data points are the elevation of the road on the centerline of the horizontal alignment, and the elevations of the road section eight feet to the left and the right of this centerline. All three elevations should be input as real numbers (i.e., 15.0) as the program input format is F7.3.

```
WHAT IS THE NEXT STATION NO.?
? 127.0
IS THIS STATION A CURVE PC?
IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
2 FOR RIGHT CURVE
? 0
WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
? 15.0
WHAT IS CORRESPONDING ELEVATION CENTERLINE?
? 15.0
WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
? 14.0
```

STEP 13A

After input for every station in the vertical alignment road section has been completed, the program outputs data on the existing conditions of the proposed road section. This output is in tabular form. From left to right this output reads as follows: station number, left side elevation, centerline elevation, right side elevation and slope. The slope outputted is the slope between the centerline elevations of adjacent stations in percentage.

STA NO	LS ELEV	CL ELEV	RS ELEV	SLOPE
127.000	15.000	15.000	14.000	0.000
128.000	15.000	15.000	14.000	0.000
129.000	16.000	16.000	16.000	1.000
129.920	16.000	16.800	18.000	.870
130.000	17.000	17.000	15.000	2.500
131.000	17.000	17.000	16.000	0.000
132.000	17.000	19.000	17.000	2.000
133.000	10.000	20.000	17.000	1.000
134.000	19.000	21.000	19.000	1.000
135.000	20.000	20.500	21.000	-.500
136.000	20.500	20.000	21.000	-.500
137.000	19.000	19.000	20.000	-1.000
138.000	18.000	18.000	19.000	-1.000
139.000	17.000	18.000	18.000	0.000
140.000	17.000	18.000	18.000	0.000
141.000	18.500	18.500	17.500	.500

STEP 14

Next the program asks the user to review the existing slopes and determine the final design grades between station centerlines. It is important at this point for the user to consider the quantities of cut and fill in his design. A design grade sufficiently different from the existing ground slope will yield a final design

centerline elevation much higher or much lower than the existing ground centerline. This in turn will cause a large quantity of cut or fill for that road section. The user should try to minimize cut and fill quantities if practical. Another point the user should consider is the rollercoaster effect. The program will design parabolic vertical curves whenever two consecutive design grades between stations are unequal. If the user tries too hard to follow the existing ground slope in choosing his design grades, every design grade chosen will be unequal to the adjoining design grades. This will yield a vertical curve at nearly every station and a very uneven road. Therefore the user must use sound engineering judgement in the selection of his design grades. The design grades should be inputted as a real number (i.e., -1.0%) as the program input format is F7.3.

YOU MUST REVIEW THE SLOPE AND DETERMINE THE
DESIRED GRADE OR GRADES.

WHAT IS THE GRADE FROM 127.00 TO 128.00
? 1.0
WHAT IS THE GRADE FROM 128.00 TO 129.00
? 1.0
WHAT IS THE GRADE FROM 129.00 TO 129.92
? 1.0
WHAT IS THE GRADE FROM 129.92 TO 130.00
? 1.0
WHAT IS THE GRADE FROM 130.00 TO 131.00
? 1.0
WHAT IS THE GRADE FROM 131.00 TO 132.00
? -0.5
WHAT IS THE GRADE FROM 132.00 TO 133.00
? -0.5
WHAT IS THE GRADE FROM 133.00 TO 134.00
? -0.5
WHAT IS THE GRADE FROM 134.00 TO 135.00
? -0.5
WHAT IS THE GRADE FROM 135.00 TO 136.00
? -0.5
WHAT IS THE GRADE FROM 136.00 TO 137.00
? -0.5
WHAT IS THE GRADE FROM 137.00 TO 139.00
? 0.0
WHAT IS THE GRADE FROM 138.00 TO 139.00
? 0.0
WHAT IS THE GRADE FROM 139.00 TO 140.00
? 0.0
WHAT IS THE GRADE FROM 140.00 TO 141.00
? 0.0

STEP 15

Next the program asks the user to input the final grade center-
line elevation of the first station in the vertical alignment
road section. This should be input as a real number (i.e., 15.0)
as the program input format is F7.3.

WHAT IS THE FINAL GRADE CENTERLINE ELEVATION
OF THE FIRST STATION ?
? 15.0

STEP 16

Finally, the program identifies the tangent intersection stations of the design grades (V), and asks the user to input the number of stations (B) the user desires in the vertical curve at each intersection station. The value of B will be the total length of the vertical curve, with half this length coming before the intersection station and half after the intersection point. This data should be input as a real number (i.e., 2.0 stations) as the program input format is F7.3.

```
A VERTICAL CURVE IS NECESSARY AT STA 131.00
HOW MANY STATIONS DO YOU WANT TO INCLUDE IN THE CURVE ?
? 2.0
```

```
A VERTICAL CURVE IS NECESSARY AT STA 137.00
HOW MANY STATIONS DO YOU WANT TO INCLUDE IN THE CURVE ?
? 4.0
```

STEP 16A

Output for both the VERTICAL ALIGNMENT and CROSS SECTION design stages is as shown below. The program automatically calculates a standard cross section based on the final design centerline elevations. In those road sections where a horizontal curve creates the need for superelevation, the program automatically adjusts the road cross sections to transition from a standard crowned road at the P.C. and P.T. points to a road section of full superelevation at the horizontal curve midpoint station. As this tabular output is rather long, the cross section data has been broken down into two schedules. The first schedule consists of the

following: station number, left backside ditch elevation, left side ditch elevation, left side shoulder elevation, left side pavement edge elevation, and the pavement centerline elevation. The second schedule provides the station number, the right side pavement edge elevation, the right side shoulder elevation, the right side ditch elevation, and the right backside ditch elevation. The ditch and backside ditch elevations will be adjusted later on in the program by the intersection point and cross section subroutines.

***** CROSS SECTION *****

STA	BACNL	DITL	SHL	PVMIL	CL
127.000	16.209	13.982	14.265	14.517	15.000
128.000	17.209	14.982	15.265	15.517	16.000
129.000	18.209	15.982	16.265	16.517	17.000
129.920	19.129	16.902	17.185	17.437	17.920
130.000	19.209	16.982	17.265	17.517	18.000
131.000	19.834	17.607	17.890	18.142	18.625
132.000	19.709	17.482	17.765	18.017	18.500
133.000	19.209	16.982	17.265	17.517	18.000
134.000	18.709	16.482	16.765	17.017	17.500
135.000	18.209	15.982	16.265	16.517	17.000
136.000	17.772	15.544	15.828	16.080	16.563
137.000	17.459	15.232	15.515	15.767	16.250
138.000	17.272	15.044	15.328	15.580	16.063
139.000	17.209	14.982	15.265	15.517	16.000
140.000	17.209	14.982	15.265	15.517	16.000
141.000	17.209	14.982	15.265	15.517	16.000

STA	PVMTR	SHR	DITR	BACKR
127.000	14.517	14.265	12.767	14.994
128.000	15.517	15.265	13.767	15.994
129.000	16.517	16.265	14.767	16.994
129.920	17.437	17.185	15.687	17.914
130.000	17.517	17.265	15.767	17.994
131.000	18.142	17.890	16.392	18.619
132.000	18.017	17.765	16.267	18.494
133.000	17.517	17.265	15.767	17.994
134.000	17.017	16.765	15.267	17.494
135.000	16.517	16.265	14.767	16.994
136.000	16.080	15.828	14.329	16.557
137.000	15.767	15.515	14.017	16.244
138.000	15.580	15.328	13.829	16.057
139.000	15.517	15.265	13.767	15.994
140.000	15.517	15.265	13.767	15.994
141.000	15.517	15.265	13.767	15.994

STEP 17

The INTERSECTION POINT design stage is computed by the subroutine INSECT. The program initializes this design stage with some comments and then requests input of the slope grade at each station for both the right and left hand side of the road. Input consists of an integer (i.e., -2) as the program input format is I2. The INSECT subroutine internally converts this data to a ratio of the rise over the run (i.e., 2 means 1:2 slope). The subroutine then compares the right/left design shoulder elevation with right/left existing ground elevation to determine whether the cross section is in cut or fill. From this calculation, the program is able to determine the toe of the slope in a fill section by applying the slope grade downward from the road shoulder elevation. If the section is found to be in cut, the program applies the slope upward from the backside ditch elevation to the top of slope.

=====

SUBROUTINE INSECT COMPUTES THE INTERSECTION OF THE
SIDESLOPES WITH THE EXISTING GROUND AT EACH STA.
INPUT A SLOPE GRADE IN RESPONSE TO THE PROGRAMS QUESTIONS.
USE ANY WHOLE NUMBER UP TO TWO DIGITS.
I.E. USE 2 FOR 1:2 OR 4 FOR 1:4

WHAT IS THE RIGHT SLOPE GRADE AT STA. 127.00?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 128.00?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 129.00?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 129.92?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 130.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 131.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 127.00?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 128.00?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 129.00?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 129.92?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 130.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 131.00?
? 2

STEP 17A

After the program has determined whether the road section is in cut or fill, the program calculates the right and left existing ground cross section slope. This is determined by calculating the difference in the right/left existing ground elevations over the horizontal distance between these elevation points (28 feet).

If the existing ground cross section slope is found to be equal to the design slope grade, the two grades will be parallel and no intersection points could be calculated. The INSECT subroutine checks this possibility, and if found, outputs an error message and returns the user to Step 17.

WHAT IS THE RIGHT SLOPE GRADE AT STA. 100.00?
? 2

THE SLOPE GRADE CHOSEN IS PARRALLEL WITH
THE EXISTING GROUND. CHOOSE ANOTHER SLOPE GRADE.

WHAT IS THE RIGHT SLOPE GRADE AT STA. 100.00?
? 1

STEP 17B

The INSECT subroutine determines the top/toe of slope by calculating the elevations of the design road section at the slope and the extrapolated existing ground, in one foot horizontal increments. At some point these two slopes will cross if they are not parallel. The intersection point of the two slopes is defined by an elevation and a distance from the centerline. This data is then output by the INSECT subroutine in the tabular format shown below.

===== INTERSECTION POINTS =====

STA NO	LEFT HAND SIDESLOPE			RIGHT HAND SIDESLOPE		
	GRADE	ELEV	DIST	GRADE	ELEV	DIST
127.00	3	15.00	34.10	3	13.93	34.10
128.00	3	15.00	37.10	3	13.89	37.10
129.00	3	16.00	37.10	3	16.00	31.10
129.92	3	15.87	40.10	3	18.91	31.10
130.00	2	17.00	34.10	2	14.72	36.10
131.00	2	17.00	34.10	2	15.89	34.10
132.00	2	16.79	34.10	2	16.86	32.10
133.00	2	4.66	58.10	2	16.79	32.10
134.00	2	19.71	30.10	2	19.49	32.10
135.00	2	20.21	32.10	2	21.99	38.10
136.00	2	20.77	34.10	2	21.56	38.10
137.00	2	19.46	32.10	2	20.24	36.10
138.00	2	18.27	30.10	2	20.06	36.10
139.00	2	18.21	30.10	2	18.99	34.10
140.00	2	18.21	30.10	2	18.99	34.10
141.00	2	19.21	32.10	2	17.99	32.10

=====

STEP 17C

The CROSS SECTIONAL AREA design stage is computed by the XSECT subroutine. No additional input is required from the user. Subroutine XSECT collects the road cross section elevation data produced in the CROSS SECTION and INTERSECTION POINT design stages and develops cross section coordinate data for each station. The program then computes the cross sectional area (in square feet) of cut or fill at each station by means of the criss-cross method. Finally the program computes volume and cumulative volume (in cubic yards) between each of the stations. The volume calculation is performed using the average end area method. The cross sectional areas, the volumes and the cumulative volumes are assigned a sign such that fill is represented by a + sign and cut is represented by a - sign. The output is presented in tabular form as shown on the following page.

=====

SUBROUTINE XSECT COMPUTES CROSS SECTIONAL AREAS

FOR EACH STA, VOLUME BETWEEN STAS, AND CUMULATIVE

VOLUME. FILL AREAS ARE GIVEN A + SIGN AND CUT AREAS

ARE INDICATED BY A - SIGN.

===== VOLUMES =====

STA NO	CROSS SECTION AREAS(SF)	VOLUME BETWEEN STATIONS(CY)	CUMULATIVE VOLUME(CY)
		0.00	0.00
127.00	-10.69	85.41	85.41
128.00	56.81	176.10	261.52
129.00	38.28	114.66	376.18
129.92	29.02	14.60	390.78
130.00	69.55	297.11	687.89
131.00	90.88	159.29	847.18
132.00	-4.87	-199.79	647.39
133.00	-103.02	-553.85	93.54
134.00	-196.05	-853.92	-760.38
135.00	-265.06	-1007.80	-1768.18
136.00	-279.15	-927.93	-2696.11
137.00	-221.93	-743.30	-3439.41
138.00	-179.45	-638.72	-4078.13
139.00	-165.46	-612.82	-4690.94
140.00	-165.46	-640.29	-5331.23
141.00	-180.30		

=====

STEP 17D

The final design stage, MASS BALANCE is computed by the MASBAL subroutine. No additional input is required from the user. Subroutine MASBAL collects the cumulative volume data produced in the CROSS SECTIONAL AREA design stage and sorts this data in descending order. The program then outputs the maximum and minimum cumulative volumes and their station locations in narrative form. This is shown below. Finally, the MASBAL subroutine routes the cumulative volume and station data to the VERSATEC plotter. A sample of this output is shown on below.

```
=====
SUBROUTINE MASBAL FINDS MAXIMUM AND MINIMUM
CUMULATIVE VOLUMES, AND IDENTIFIES THEIR STATION
LOCATION POINTS. IT ALSO PLOTS A MASS DIAGRAM ON
THE VERSATEC PLOTTER, NO INPUT IS NECESSARY.
```

```
THE MAXIMUM CUMULATIVE VOLUME IS      847.18 CUBIC YARDS
THIS OCCURS AT STATION 127.00
```

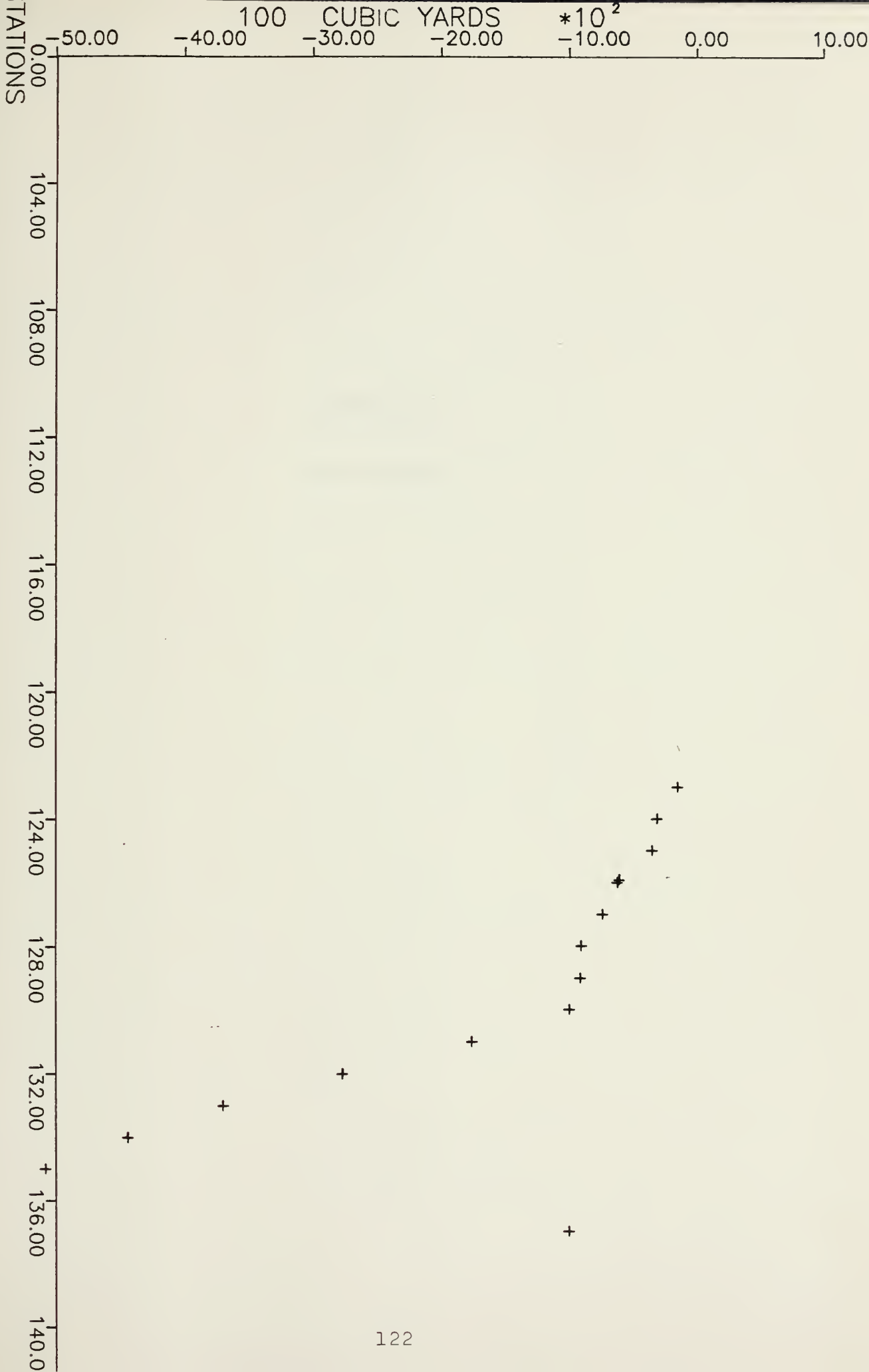
```
THE MINIMUM CUMULATIVE VOLUME IS  -5331.23 CUBIC YARDS
THIS OCCURS AT STATION 141.00
```

```
GO TO THE PLOTTER BIN AND REMOVE OUTPUT (MASS DIAGRAM)
```

```
=====
```

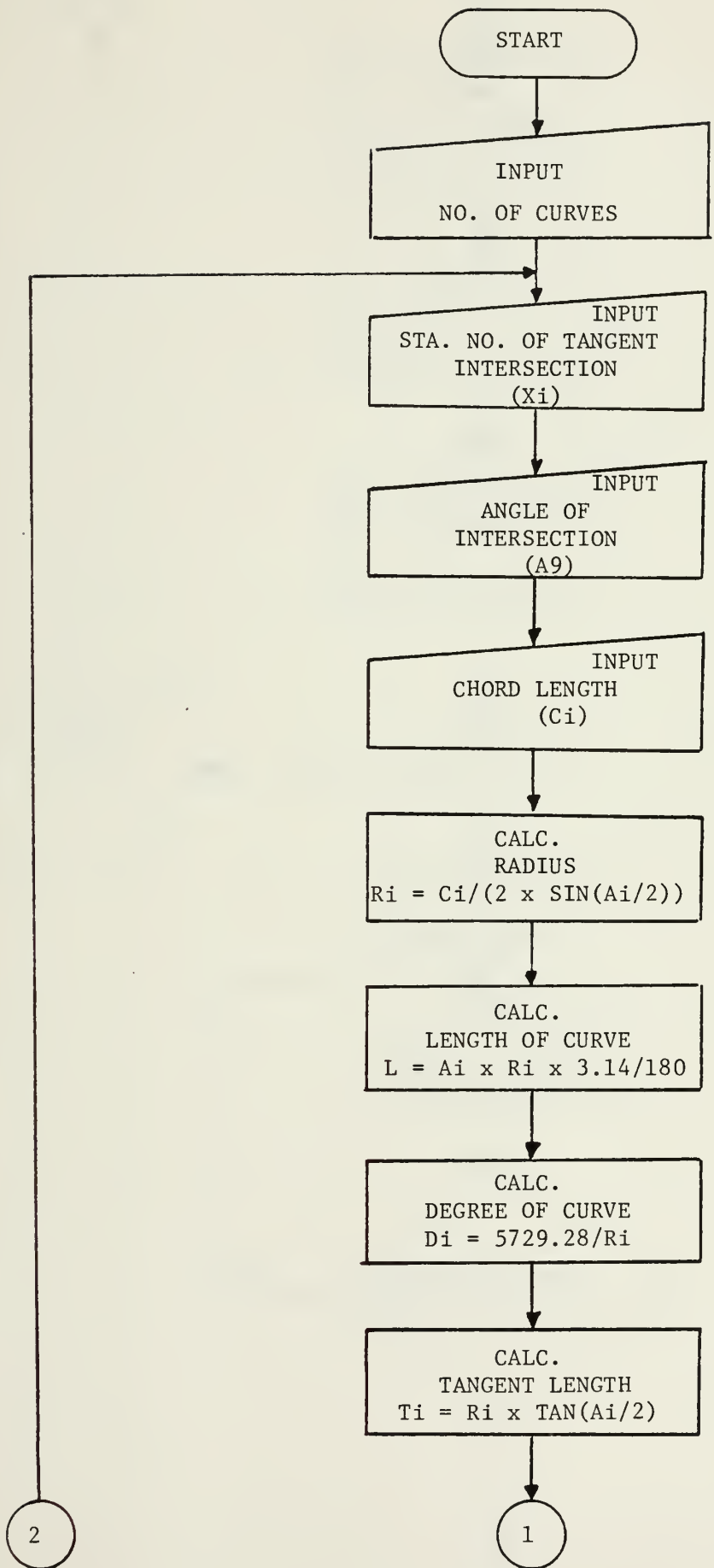
```
*****
THIS PROGRAM WILL END
      3.232 CP SECONDS EXECUTION TIME.
/PLOTCV(TAPE9)
 PLOT.    81/11/03. 12.21.35.
/bye

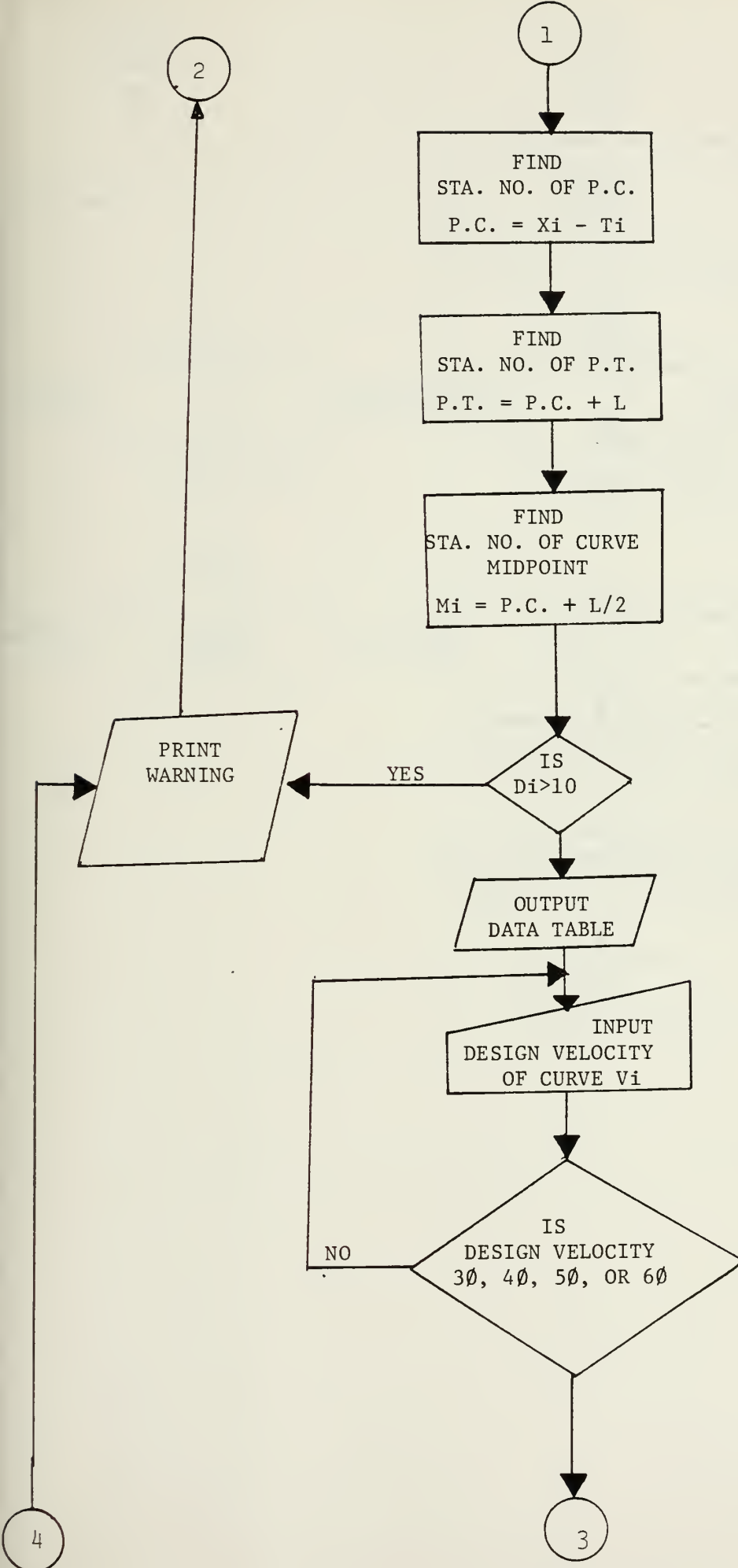
CES13AA  LOG OFF    12.23.08.
CES13AA  SRU       54.444 UNTS.
```

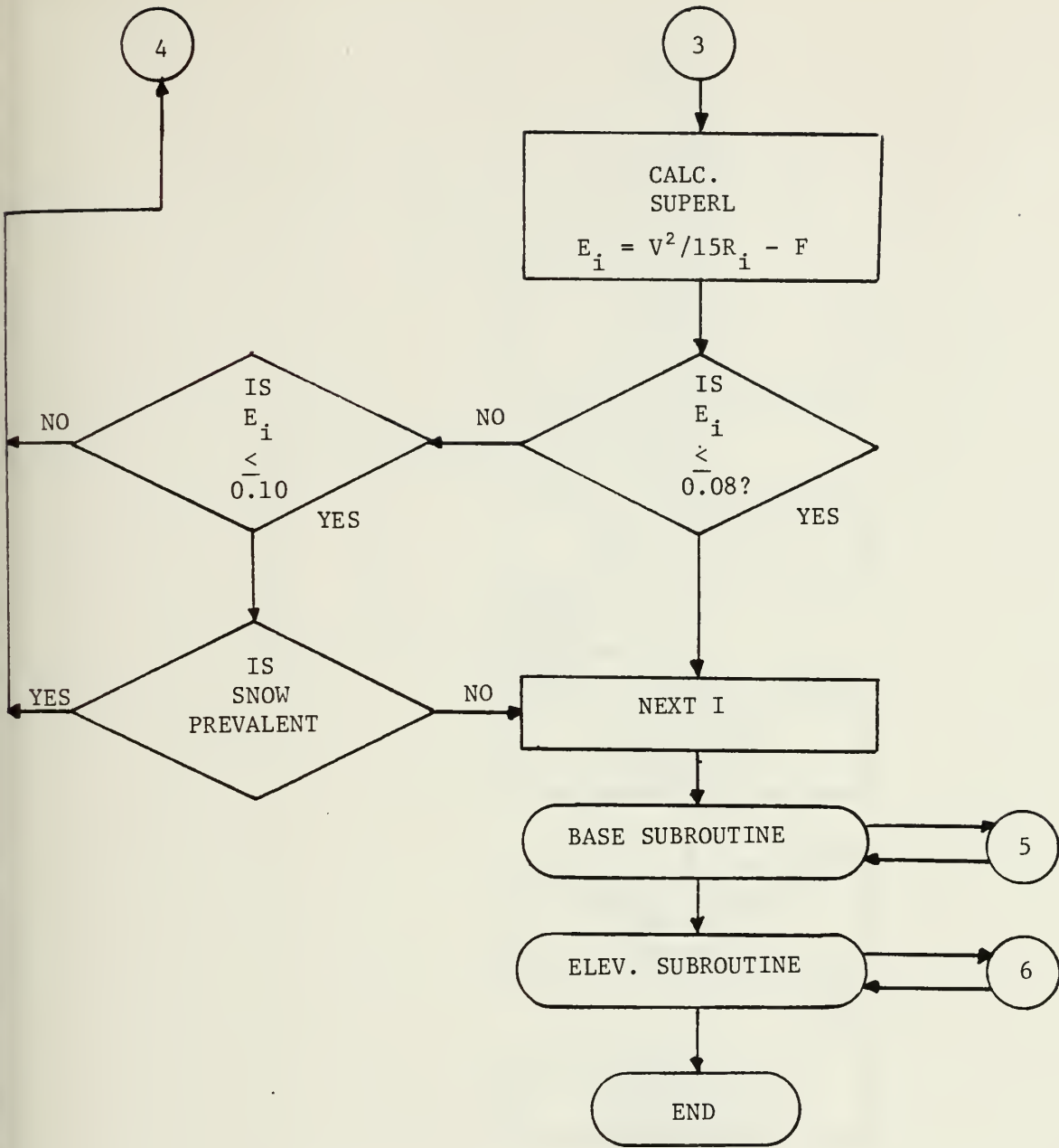


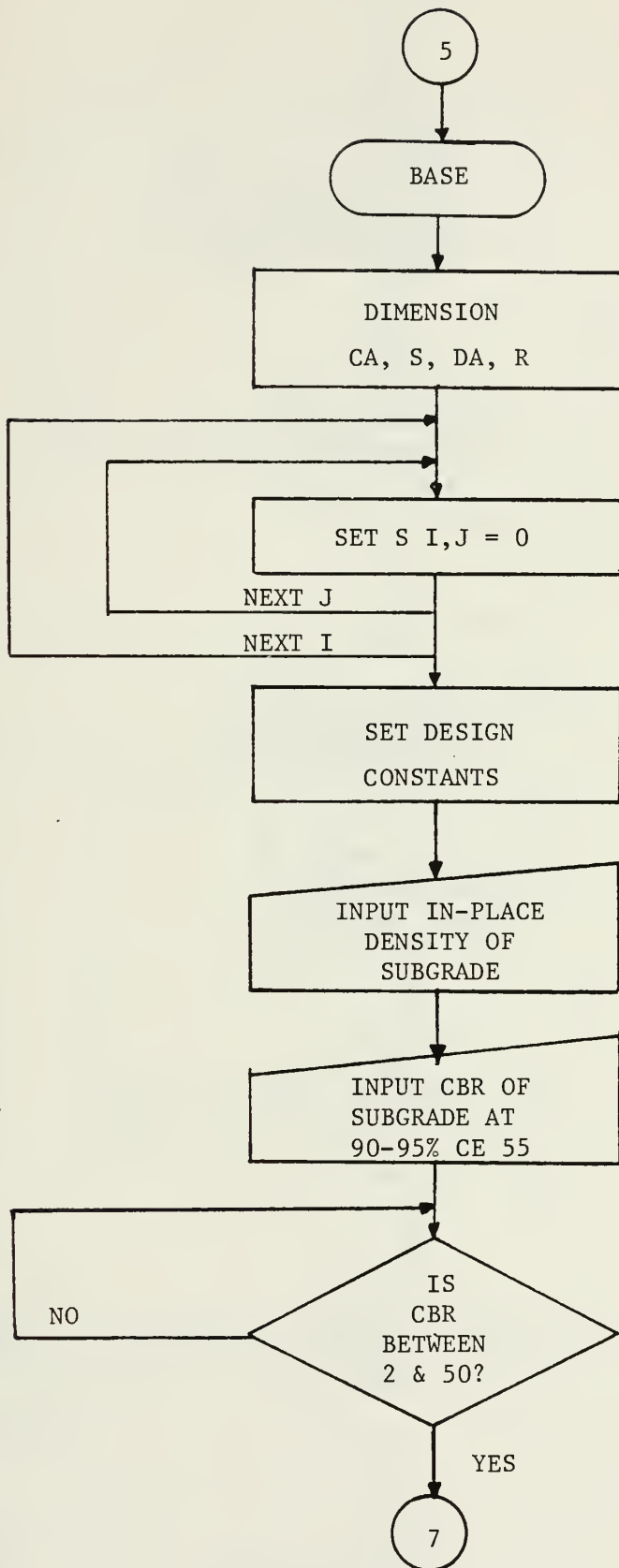
APPENDIX D

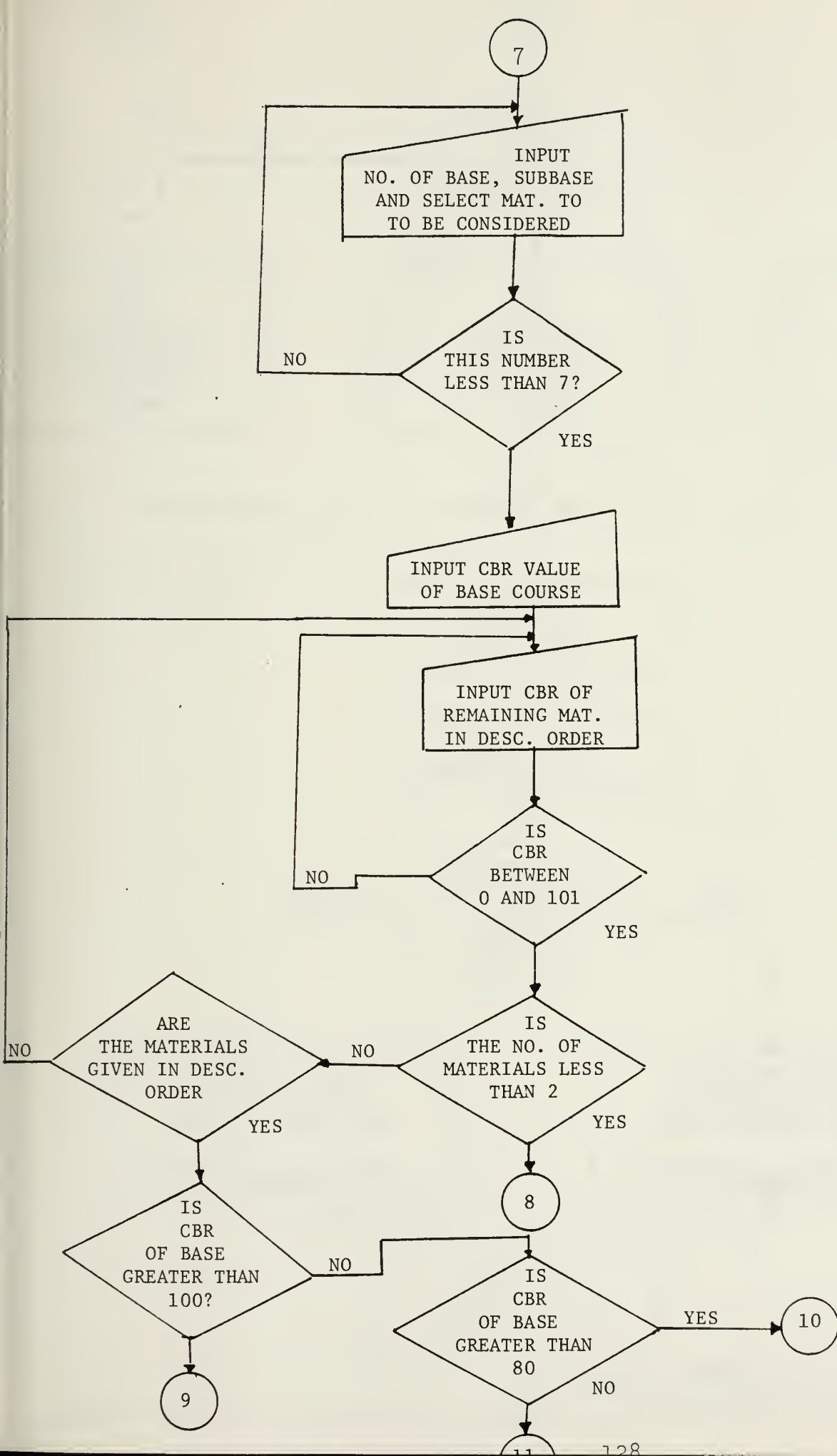
FLOW CHART

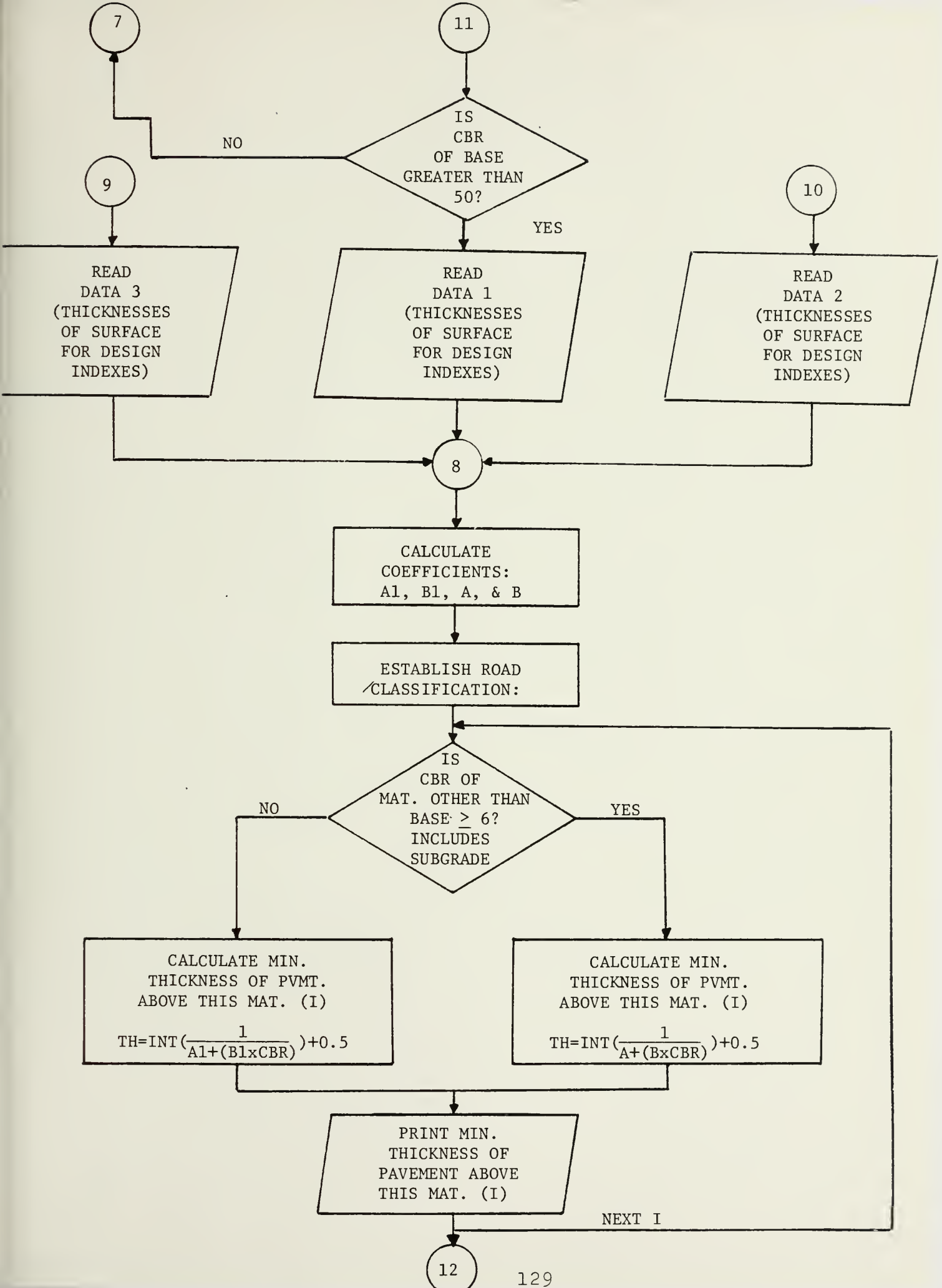












12

IS
MIN. SURFACE
THICK = 0.5 AND
CBR BASE \geq 80

YES

NO

IS
MIN. SURF.
THICK = 0.5 AND
CBR BASE \geq 100?

YES

PRINT: "BITUM.
SURFACE TREATMENT
REQUIRED" - MIN.
BASE THICKNESS
= 4 INCHES"

NO

IS
MIN. SURFACE
THICK = 1.0?

YES

PRINT: MULT.
BITUM. SURFACE
TREATMENT
REQUIRED

NO

SET X = MIN. SURF.
THICKNESS

SET Y = 4 INCHES
(ABS. MIN. FOR
BASE COURSE)

IS
 $X + Y \geq 6$

YES

NO

SET BASE THICKNESS =
6 - SURFACE THICKNESS

13

13

LET DESIGN THICKNESS
THUS FAR =
SURFACE THICKNESS +
BASE THICKNESS

LET THICKNESS OF THIS
MAT. (I) = MIN. THICKNESS
ABOVE NEXT MAT. (I) -
DESIGN THICKNESS
THUS FAR

IS
THICKNESS
OF MAT. (I)
 ≥ 4

YES

NO

SET THICKNESS MAT(I)
= 4

LET DESIGN THICKNESS
THUS FAR = PREV.
DESIGN THICKNESS +
THICKNESS MAT. (I)

IS
NEXT
MATERIAL
SUBGRADE
?

NEXT I

NO

YES

LET THICKNESS
ABOVE SUBGRADE =
DESIGN THICKNESS
THUS FAR

14

14

PRINT DESIGN PROFILE
WITH
CBR. OF MAT.
THICKNESS OF MAT.
CUM. THICKNESS
SUBGRADE CBR

INPUT
SUBGRADE COHESION
INDICATOR

SET BASE COMPACT.
= 100%

CALC. COEFF. FOR
MIN. THICKNESS BASED ON
COMPACTION

FOR
MAT. BELOW BASE
IS CBR OF MAT. (I)
< 20?.

YES

NO

YES

COMPACTION
MAT. (I)=90%

COMPACTION
MAT. (I)=100%

SET COMPACTION
MAT. (I)=95%

NEXT I

15

15

CALC. THICKNESS ABOVE
SUBGRADE BASED ON
COMPACTION I = 100,
95, 90, 85, or 80
FOR COHESIVE SUBGRADE
THICKNESS = INT. (A1+(B1xI)),
THICKNESS = INT. (A+(BxI))
OTHERWISE

NEXT I

PRINT:
COMPACTION
PROFILE

IS
FILL
COMPACTION
≥ 80 OR THE
DESIGN THICKNESS = MIN.
CALC. FOR 80% FILL
COMPACTION

ADD 4" TO DESIGN
THICKNESS SUBGRADE
MAT. AT 80% COMP.

FILL = SUBGRADE
MAT. AT FILL
COMPACTION CALC.

FILL = SUBGRADE
MAT. AT COMPACTION
90 or 95 DEPEND. ON
COHESION

16

16

DOES
FILL COMPACTION
EXCEED INPLACE DENS.
OF SUBGRADE
BY > 10%

YES

NO

SCARIFY SUBGRADE
TO A DEPTH OF 6"
AND COMPACTION AT
INPLACE DENS. + 5%

PLACE FILL ON
SUBGRADE AT
IN-PLACE DENSITY

PRINT:
FILL PROFILE

CUT SECTIONS

IS
DESIGN
THICKNESS > MIN. REQD.
FOR 100% COMPACTION

NO

YES

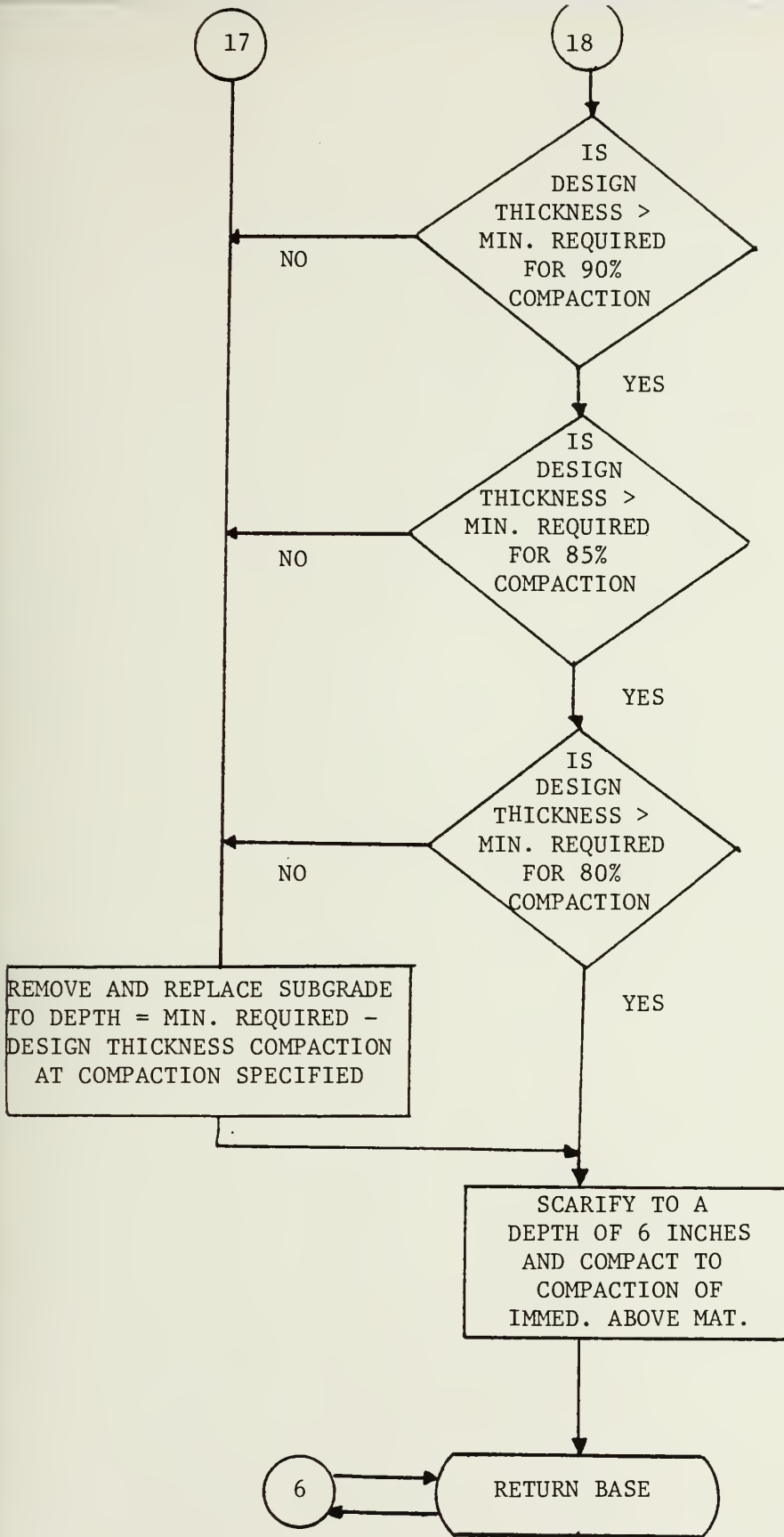
IS
DESIGN
THICKNESS > MIN. REQD.
FOR 95% COMPACTION

NO

YES

17

18



ELEV.

INPUT:
STATION
ELEVATIONS

CALC. SLOPE
OF CENTERLINE
STA. I TO I + 1

NEXT I

OUTPUT:
SLOPE AND
ELEVATIONS

INPUT
GRADE STA. I TO
I + 1

NEXT I

DOES A
GRADE BREAK
OCCUR?

YES

INPUT
LENGTH OF
VERTICAL CURVE

CALC., VERTICAL
CURVE ELEVATION

NO

19

IS THIS STATION A.P.C.

NO

YES

IS THE CURVE RIGHT OR LEFT HAND

RIGHT

LEFT

APPLY RIGHT SUPERELEVATION AND CALC. CROSS SECTION

APPLY LEFT SUPERELEVATION AND CALC. CROSS SECTION

CALC. CROSS SECTION

NEXT STATION

NEXT STATION

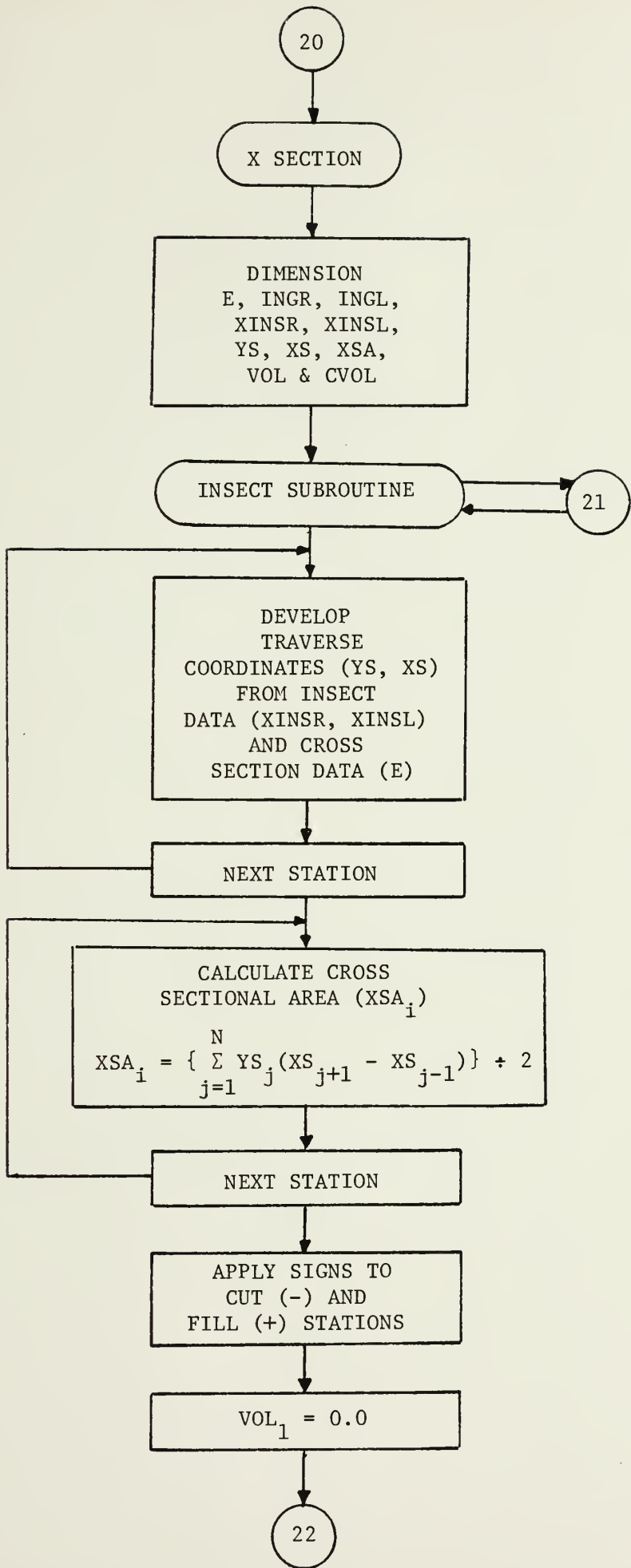
OUTPUT: VERTICAL ALIGNMENT AND CROSS SECTION DATA

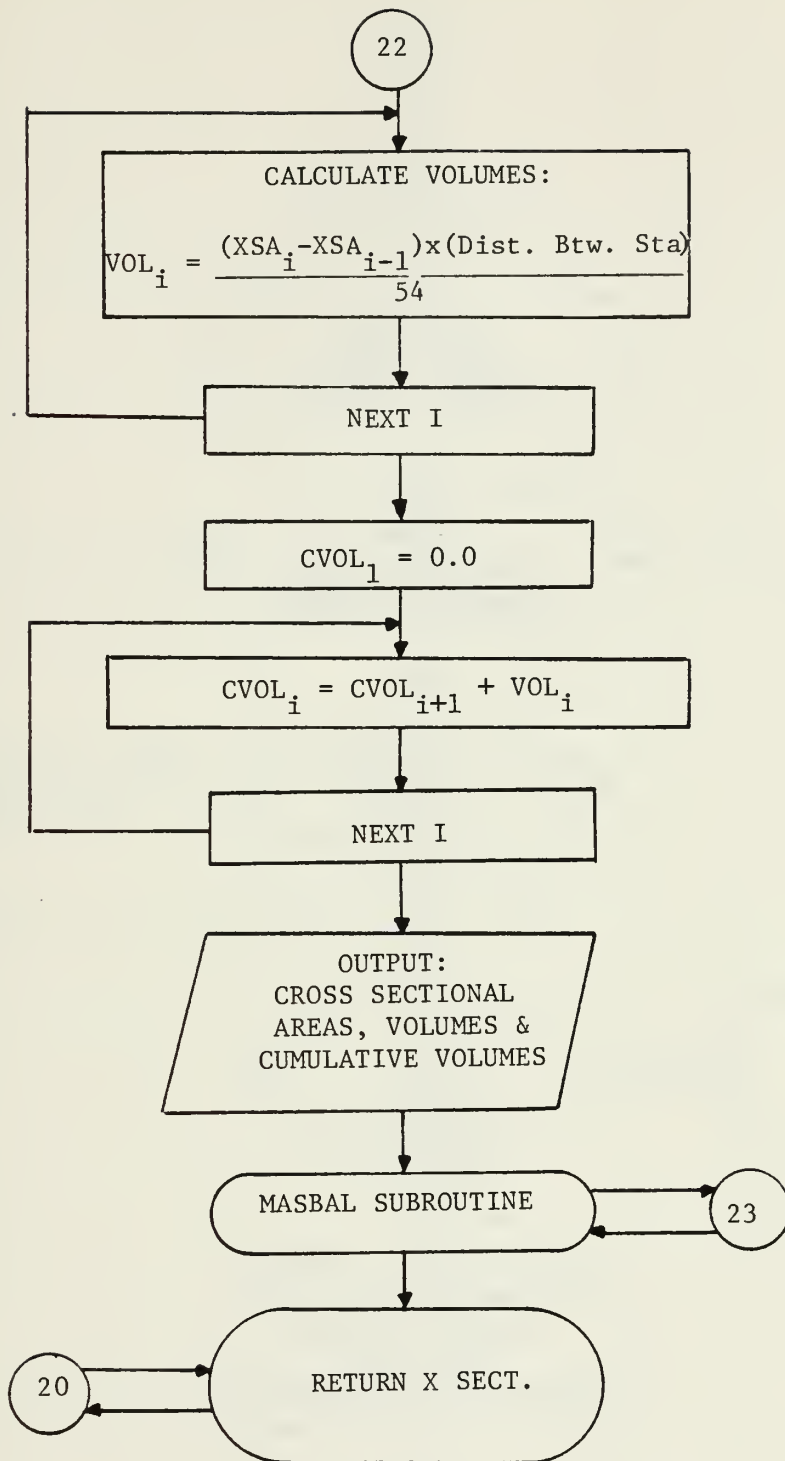
X SECTION SUBROUTINE

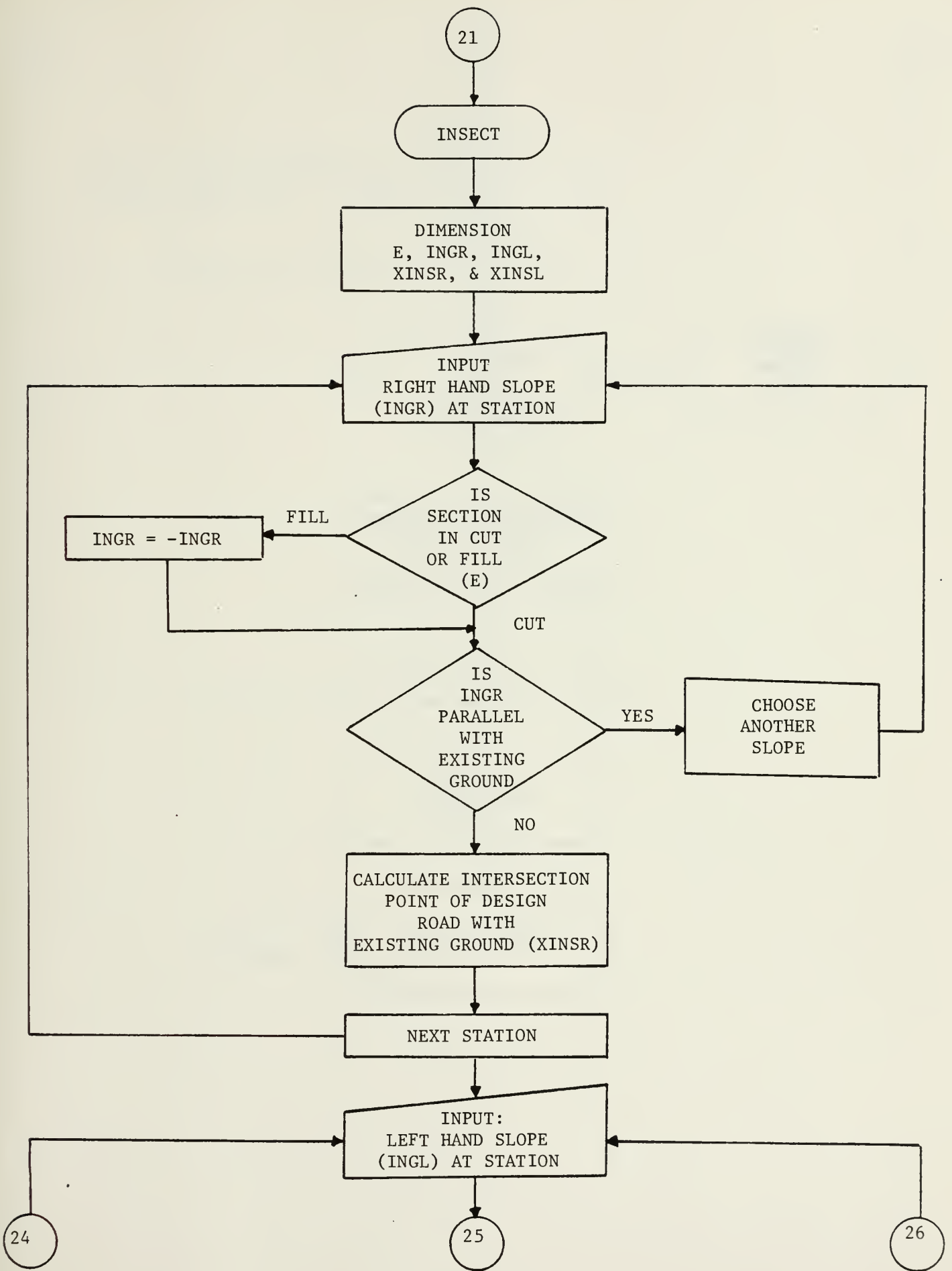
20

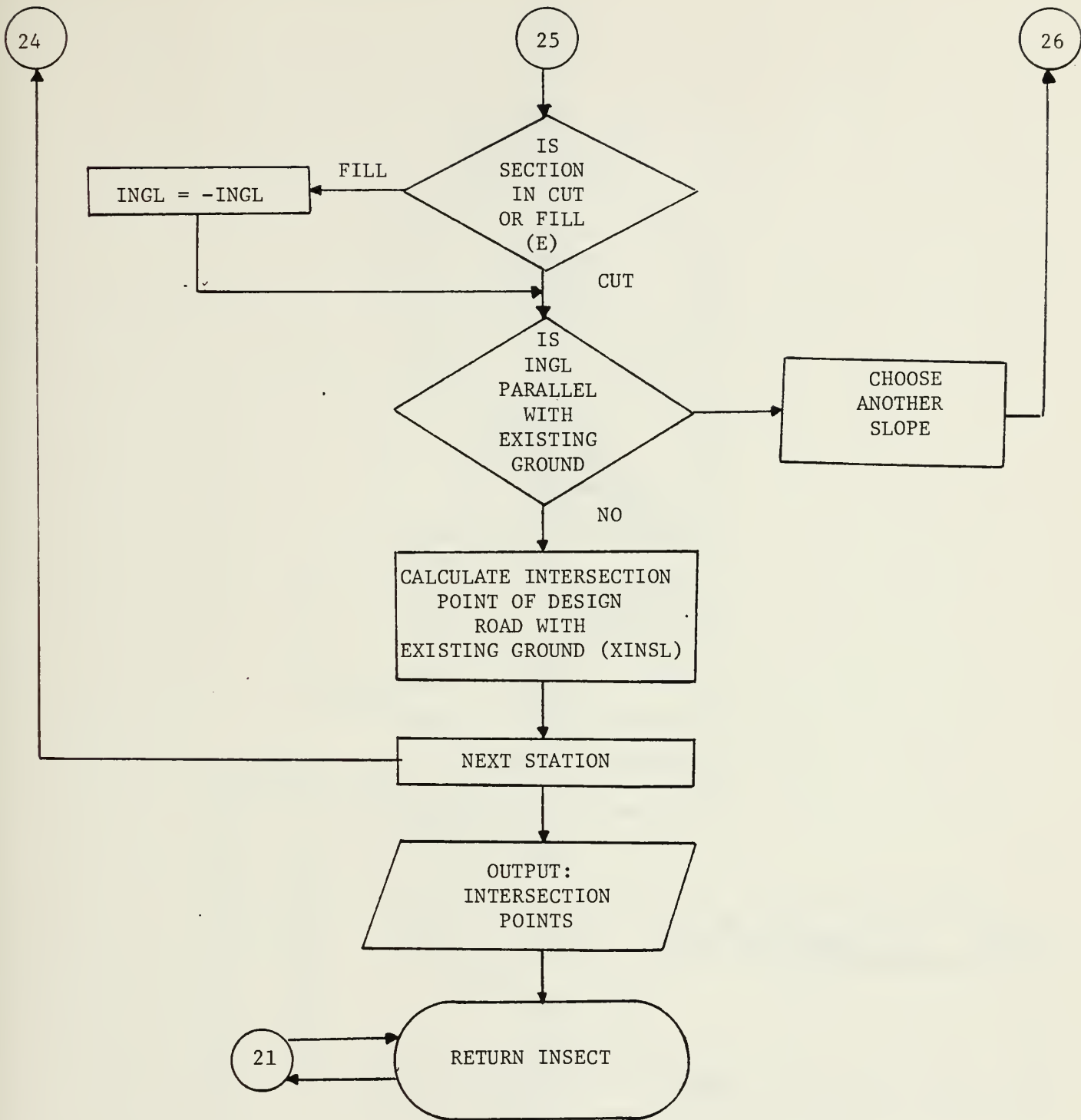
RETURN ELEV.

6









MASBAL

DIMENSION
E, CVOL,
X, Y, YMAX

DEFINE
COORDINATES
 $X_i = \text{STA. (E)}$
 $Y_i = \text{CUMULATIVE}$
 VOLUME (CVOL)

$$Y_i = Y_i + 1,000,000$$

NEXT

$$\text{AMAX} = Y_i$$

IS
 $Y_i < \text{AMAX}$

NO

$$\text{AMAX} = Y_i$$

YES

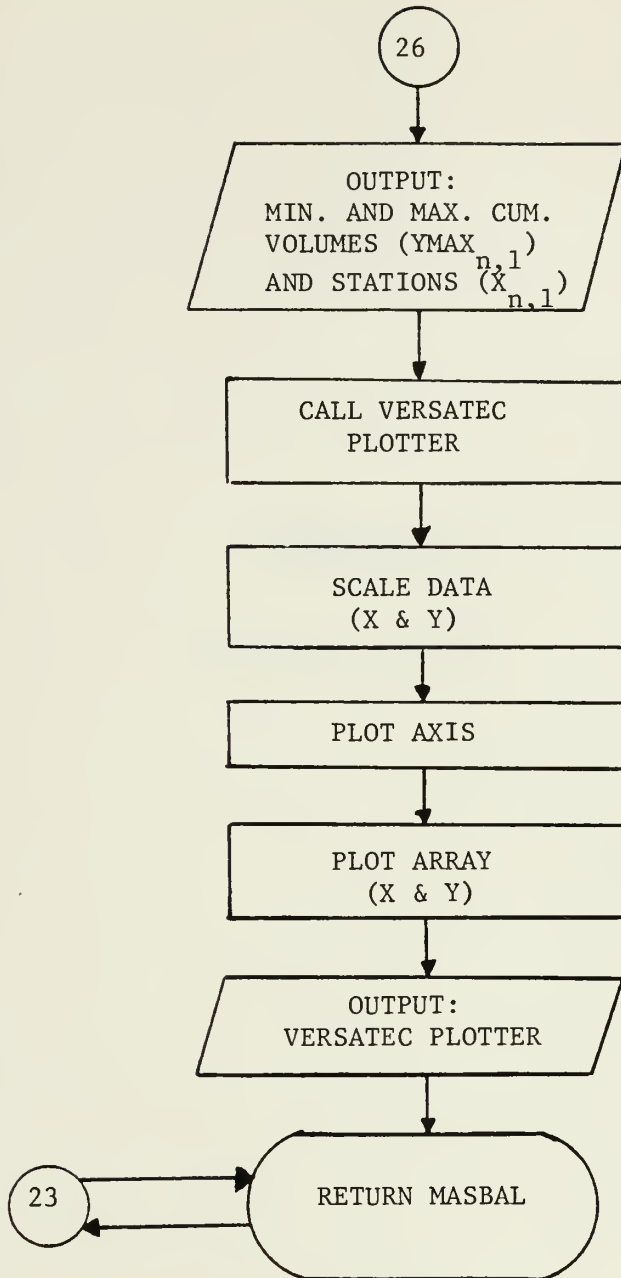
NEXT I

$$\text{YMAX}_i = \text{AMAX}$$

NEXT I

$$Y_i := \text{YMAX}_i - 1,000,000$$

NEXT I



APPENDIX E

PROGRAM LISTING


```

      OLD,FIRD
      /ATTACH,CALCOMP/UN=LIBRARY
      /LIBRARY(CALCOMP)
      /LIBRARY(CALCOMP)
      /FTN,I=FIRD,R=0,TS
1 FTN 4:8+528      81/11/03, 00:30:59 PAGE 1
      PROGRAM MAIN      73/74 TS

      PROGRAM MAIN(INPUT,OUTPUT,TAPES=INPUT,TAPE6=OUTPUT,TAPE9
      >>>> )
      C
      C BEGIN FIRD, ROAD DESIGN PROGRAM MODELLED AFTER
      C THE SPECIAL RESEARCH PROJECT OF R. SETZEKORN
      C
      C DIMENSION H(20,8),R(20,4)
      C DIMENSION S(10,6)
      C DIMENSION G(50),E(50,15)
      C INTEGER U
      C INTEGER V
      C INTEGER W
      C INTEGER D1
      C
      C JCL COMMANDS
      C OLD,FIRD
      C
      C 15 C ATTACH,CALCOMP/UN=LIBRARY
      C LIBRARY(CALCOMP)
      C FTN,I=FIRD,L=0,TS
      C LGO

```



```

C          PLOTCV(TAPE9)
20 C
101      WRITE(6,101)
        FORMAT(//,14X,33('*'), ' ROAD DESIGN ',33('*'),//)
102      WRITE(6,102)
        FORMAT('THIS PROGRAM IS A SERIES OF SUBROUTINES DESIGNED
>>>>
>>>> 25      C*TO ASSIST THE ENGINEER IN A MILITARY ROAD DESIGN.',/,
        C*THE OPERATION OF THE PROGRAM IS INTERACTIVE.',/,
        C*AND REQUIRES THE USE OF THE USERS MANUAL.',//)
        WRITE(6,103)
        FORMAT(44X,'HORIZONTAL ALIGNMENT',//)
30      WRITE(6,104)
        FORMAT('ENTER NO. OF CURVES THIS SECTION')
105      READ(5,105)N
        FORMAT(I2)
35 70      IF(N.EQ.0) GO TO 175
75      DO 106 I=1,N
        WRITE(6,107)I
107      FORMAT(//,'ENTER STA NO OF TANGENT INTERSECTION OF CURVE
>>>>
>>>>          ',I2)
40      READ(5,108)H(I,1)
        FORMAT(F7,3)
        WRITE(6,109)
        FORMAT(//,'ENTER ANGLE OF INTERSECTION OF CURVE IN DEGRE
>>>>          ES')
45      READ(5,110)A
        FORMAT(F7,3)
        X=(A*6.28318)/360.
        WRITE(6,111)
        FORMAT(//,'ENTER CHORD LENGTH IN FEET')
112      READ(6,112)C
        FORMAT(F7,3)
        WRITE(6,113)

```



```

50 113      FORMAT(//)
           H(I,2)=C/(2.*SIN(X/2.))
           H(I,3)=A#H(I,2)*3.1416/180.
           H(I,4)=5729.28/H(I,3)
1. FTN 4.8+528      81/11/03. 09.30.59 PAGE 2
PROGRAM MAIN      73/74  TS

           H(I,5)=H(I,2)*TAN(X/2.)
           H(I,6)=H(I,1)-(H(I,5)/100.)
           H(I,7)=H(I,1)+(H(I,3)/100.)
           H(I,1)=H(I,6)+(H(I,3)/200.)
           IF(H(I,4).GT.10.) GO TO 135
           CONTINUE
           GO TO 145
           60
           135 WRITE(6,114)
           114 FORMAT("WARNING D TOO HIGH REESTABLISH PC, FT, ANDC",//)
           GO TO 75
           145 WRITE(6,115)
           115 FORMAT(T10,"STA NO.",T21,"RAD",T30,"L OF C",T45,"D OF C",
           >>>> T60,"TAN
           CLEN",T70,"PC STA",T80,"FT STA",/)
           DO 116 I=1,N
           WRITE(6,117)H(I,1),H(I,2),H(I,3),H(I,4),H(I,5),H(I,6),H(
           >>>> I,7)
           117 FORMAT(T9,F7.3,T20,F7.2,T30,F7.3,T45,F7.5,T60,F7.3,T70,F
           >>>> 7.3,T80,F7
           >>>> C,3)
           70
           116 CONTINUE
           U=N
           GO TO 205
           C SUBROUTINE 1400
           75 175 CALL BASE (S,N)
           C SUBROUTINE 4000

```



```

180 CALL ELEV(E,G)
WRITE(6,120)
FORMAT("THIS PROGRAM WILL END")
80 GO TO 9999
205 WRITE(6,121)
121 FORMAT(//,37X," SUPERELEVATION SUPERL")
DO 122 I=1,N
220 WRITE(6,123)I
85 123 FORMAT(//,"WHAT IS DESIGN VELOCITY OF CURVE ",I2)
READ(5,124)V
124 FORMAT(I2)
IF(V.EQ.30) GO TO 260
IF(V.EQ.40) GO TO 270
IF(V.EQ.50) GO TO 280
IF(V.EQ.60) GO TO 290
WRITE(6,125)
125 FORMAT(//,"INVALID VELOCITY--TRY AGAIN")
GO TO 220
95 260 F=0.16
270 GO TO 295
F=0.1
80 TO 295
280 F=0.14
GO TO 295
100 290 F=0.13
295 X=V
H(I,8)=(X**2/(15.*H(I,2)))-F

```



```

105          IF(H(I,8).LE.0.) GO TO 350
            IF(H(I,8).LE.0.08) GO TO 122
            IF(H(I,8).LE.0.10) GO TO 325
            WRITE(6,126)I
1 FTN 4.8+528      81/11/03, 09.30.59 PAGE    3
PROGRAM MAIN      73/74    TS

110          126  FORMAT(/,,"WARNING-RADIUS MUST INCREASE TO DECREASE SUPE
>>>>  RL OF CURV
            CE",I2)
            GO TO 135
127          325  WRITE(6,127)
            127  FORMAT("WARNING-IS SNOW PREVALENT?",/,
C"ENTER 1 IF YES AND 2 IF NO.",/,
C/,)
115          115  READ(5,128)W
            128  FORMAT(I1)
            IF(W.EQ.1) GO TO 70
            IF(W.EQ.2) GO TO 122
            H(I,8)=0.
120          120  CONTINUE
            129  WRITE(6,129)
            129  FORMAT(T10,"STA NO",T21,"RAD",T30,"L OF C",T45,"SUPERL",
T60,"TAN L
            CEN",T70,"PC STA",T80,"PT STA",/)
            DO 130 I=1,N
125          125  WRITE(6,131)H(I,1),H(I,2),H(I,3),H(I,8),H(I,5),H(I,6),H(I
>>>>  I,7)
131          131  FORMAT(T9,F7.3,T21,F7.2,T30,F7.3,T43,F10.8,T60,F7.3,T70,
>>>>  F7.3,T80,F
            C7.3)
130          130  CONTINUE
            GO TO 175
130          130 9999  JOE=1
            END
420008 CM STORAGE USED      1.225 SECONDS
1 FTN 4.8+528      81/11/03, 09.30.59 PAGE    1

```


SUBROUTINE BASE 73/74 TS

```
C
C
C
5
10
132
134
133
15
136
1445
137
20
138
25
1460
139
N.
>>>>
1470
30
557.
SUBROUTINE BASE(S,N)
DIMENSION S(10,6),R(20,4)
INTEGER DI
WRITE(6,132)
FORMAT(//)
DO 133 I=1,6
DO 134 J=1,10
S(J,I)=0,
CONTINUE
CONTINUE
WRITE(6,141)
FORMAT("WHAT DESIGN INDEX IS TO BE USED?")
READ(5,136)DI
FORMAT(I1)
WRITE(6,137)
FORMAT("WHAT IS THE IN-PLACE (ZCESS) OF THE SUBGRADE?")
READ(5,138)ARCS
FORMAT(F4,1)
S(10,2)=ARCS
IF(S(10,2).LT.20.) GO TO 1460
IF(S(10,2).GT.100.) GO TO 1460
GO TO 1470
WRITE(6,139)
FORMAT("INPLACE DENSITY MUST BE GIVEN AS ZCESS, TRY AGAIN")
GO TO 1445
WRITE(6,9941)
FORMAT("WHAT IS THE CBR OF THE SUBGRADE AT 90% TO 95% CE")
>>>>
```



```

140 READ(5,140)BCDS
    FORMAT(F4,1)
    S(10,1)=BCDS
    IF(S(10,1).LT.1.) GO TO 1485
    IF(S(10,1).GT.100.) GO TO 1485
35   GO TO 1495
    1485 WRITE(6,1000)
    1000 FORMAT("VALUES FOR CBR SHOULD RANGE FROM ABOUT 2 TO 50.
>>>> TRY AGAIN.
        C")
40   GO TO 1470
    1495 WRITE(6,1001)
    1001 FORMAT("HOW MANY TYPES OF BASE, SUBBASE, AND SELECT MATE
>>>> RIALS"//)
        C"DO YOU WISH TO CONSIDER?( DO NOT INCLUDE SUBGRADE OR PA
>>>> VT)"/)
45 1002 READ(5,1002)L
    FORMAT(I1)
    IF(L.EQ.0) GO TO 1515
    IF(L.GT.7) GO TO 1515
    GO TO 1530
50 1003 WRITE(6,1003)
    FORMAT("THIS PROGRAM WILL HANDLE UP TO 7 DIFFERENT SOIL
>>>> "/)
        C"THIS SHOULD BE MORE THAN SUFFICIENT.")
1 FTN 4.2+528 81/11/03, 09.30.59 PAGE 2
SUBROUTINE BASE 73/74 TS
    GO TO 1495
1530 WRITE(6,1004)
1004 FORMAT("WHAT IS THE CBR VALUE OF THE BASE COURSE?")
55 READ(5,1005)CDES

```



```

1005 FORMAT(F4.1)
      S(2,1)=CPES
      LCUCH=L+1
      DO 1006 I=3,LCUCH
60 1545 WRITE(6,1007)
1007 FORMAT("WHAT IS THE NEXT HIGHEST CBR?")
      READ(5,1008)DEFS
1008 FORMAT(F5.1)
      S(I,1)=DEFS
      IF(S(I,1).LE.0.) GO TO 1560
      IF(S(I,1).GE.101.) GO TO 1560
      GO TO 1006
1560 WRITE(6,1010)
1010 FORMAT("VALUES FOR CBR SHOULD RANGE FROM ABOUT 2 TO 100.
      >>>> " / )
      C"TRY AGAIN.")
      GO TO 1545
1006 CONTINUE
      IF(L.LT.2) GO TO 1665
      DO 1011 I=2,LCUCH
75 IF(S(I+1,1).LE.S(I,1)) GO TO 1011
      WRITE(6,1012)
1012 FORMAT("CBR VALUES WERE NOT GIVEN IN DESCENDING ORDER.")
      >>>> /
      C"TRY AGAIN.")

```



```

80 1011      GO TO 1530
              CONTINUE
              IF(S(2,1).GE.100.) GO TO 1630
              IF(S(2,1).GE.80.) GO TO 1640
              IF(S(2,1).GE.50.) GO TO 1645
              WRITE(6,1013)
85 1013      FORMAT('CRR OF BASE COURSE MUST BE ABOVE 50. TRY AGAIN.')
          >>>>
          )
1630        GO TO 1530
              S(1,6)=0.5
              S(2,6)=1.0
              S(3,6)=1.5
              S(4,6)=1.5
              S(5,6)=2.0
              S(6,6)=2.5
              S(7,6)=2.5
              S(8,6)=3.0
              S(9,6)=3.5
              S(10,6)=3.5
              GO TO 1665
1640        S(1,6)=0.5
              S(2,6)=1.5
              S(3,6)=1.5
              S(4,6)=2.0
              S(5,6)=2.5
              S(6,6)=3.0
              S(7,6)=3.0
              S(8,6)=3.5
100         S(9,6)=3.5
              S(10,6)=3.5
105         S(1,6)=3.5
              S(2,6)=4.0
              S(3,6)=4.0
              S(4,6)=4.0
              S(5,6)=4.0
              S(6,6)=4.0
              S(7,6)=4.0
              S(8,6)=4.0
              S(9,6)=4.0
              S(10,6)=4.0
              GO TO 1665
1645        S(1,6)=2.0
              S(2,6)=2.5
110         S(3,6)=2.5
              S(4,6)=2.5
              S(5,6)=2.5
              S(6,6)=2.5
              S(7,6)=2.5
              S(8,6)=2.5
              S(9,6)=2.5
              S(10,6)=2.5

```

J FTN 4.8+528 81/11/03. 09.30.59 PAGE 3
 SUBROUTINE ROSE 73/74 TS


```

115 S(3,6)=2.5
    S(4,6)=3.0
    S(5,6)=3.5
    S(6,6)=4.0
    S(7,6)=4.5
    S(8,6)=5.0
    S(9,6)=5.5
    S(10,6)=6.0
1665 A=1./((26.7561+4.90148*XD1)
120 A1=1./((33.4189+6.9253*XD1)
    R=1./((73.9301+16.9388*XD1)
    B1=1./((66.5999+14.6764*XD1)
    WRITE(6,1681)
1681 FORMAT("WHAT CLASSIFICATION IS TO BE USED?"/)
125 C*A=1,B=2,C=3,D=4,E=5,F=6")
    READ(5,1683)C2
1683 FORMAT(I1)
    WRITE(6,1690)
1690 FORMAT(//)
130 DO 1730 I=1,8
    J=1-I
    IF(S(J,1),EQ,0.) GO TO 1730
    IF(S(J,1),GT,6.) GO TO 1720
    I00=(1./((A1+R1)*S(J,1)))+0.5)
    S(J,3)=I00
    GO TO 1725
1720 IBB=(1./((A+R)*S(J,1)))+0.5)
    S(J,3)=IBB
1725 WRITE(6,1726)S(J,1),S(J,3)
1726 FORMAT("MIN THICKNESS ROD ABOVE SOIL WITH CBR=",F4.1," I
    S ",F4.1)
    >>>>
1730 CONTINUE
    IF(S(D1,6),EQ,0.5,AND,S(2,1),EQ,80.) GO TO 1770
    IF(S(D1,6),EQ,0.5,AND,S(2,1),EQ,100.) GO TO 1780
    IF(S(D1,6),EQ,1.) GO TO 1770
145 S(1,5)=S(D1,6)

```



```

S(2,5)=4.0
WRITE(6,1760)S(1,5)
1760 FORMAT("THE MIN REQD THICKNESS OF PAVT IS",F4.1)
GO TO 1785
150 1770 WRITE(6,1771)
1771 FORMAT("MULTIPLE BITUMINOUS SURFACE TREATMENT IS REQUIRE
>>>> D* )
GO TO 1785
1780 WRITE(6,1781)
1781 FORMAT("BITUMINOUS SURFACE TREATMENT IS REQUIRED")
155 1785 WRITE(6,1786)
1786 FORMAT("MIN REQD THICKNESS OF BASE COURSE IS 4 INCHES")
ACUCH=S(1,5)+S(2,5)
IF(ACUCH.GE.6.0.OR.C2.GT.4.0) GO TO 1815
S(2,5)=6.0-S(01,6)
160 WRITE(6,1800)S(2,5)
      81/11/03, 09.30.59 PAGE 4
SUBROUTINE BASE 73/74 TS

1800 FORMAT("MIN TOTAL THICKNESS OF PAVT PLUS BASE COURSE FOR
>>>> " ,/,
>>>> D" ,/,
C"CLASS C ROADS IS 6 INCHES. BASE COURSE WILL BE INCREASE
C"TO",F4.1,"INCHES")
1815 S(2,4)=S(1,5)
S5=S(1,5)+S(2,5)
165 DO 1890 I=3,10
JCUCH=I+1
IF(JCUCH.EQ.11) GO TO 1840
DO 1840 J=JCUCH,10
170 IF(S(J,3).GT.0.0.OR.S(J,3).LT.0.0) GO TO 1860
CONTINUE
IF(S5.GE.S(I,3)) GO TO 1860
S5=S(I,3)
S(I-1,5)=S(I,3)-S(I-1,4)

```



```

175 1860 S(I,5)=S(J,3)-S5
      IF(S(I,5).GE.4.0) GO TO 1875
      S(I,5)=4.0
1875 S(I,4)=S5
      S5=S5+S(I,5)
1880 IF(J.EQ.10) GO TO 1895
      CONTINUE
1895 S(10,4)=S5
      WRITE(6,1905)
1905 FORMAT(60('*'),/)
      WRITE(6,1915)
1915 FORMAT(22X,'DESIGN PROFILE',/,
          C', DEPTH',/,
          C'(INCHES)',/,
          C'4X',0',12X,7('='),1X,'TOP OF PAVEMENT',1X,7('='))
190 DO 2080 I=1,9
      IF(S(I,5).EQ.0.0) GO TO 2085
      L=(S(I,5)*0.5)
      DO 1960 J=1,L
195 WRITE(6,1955)
1960 FORMAT(/)
      CONTINUE
      IF(I.GT.3) GO TO 2020
      IF(I.EQ.1) GO TO 1975
      IF(I.EQ.2) GO TO 2000
      IF(I.EQ.3) GO TO 2010
200 IF(S(I,5).GT.1.) GO TO 1990
      IF(I.GT.1) GO TO 2040
      WRITE(6,8500)S(I,5)
      FORMAT(T16,F4.1,' INCHES:', SURFACE TREATMENT')
205 GO TO 2045
2040 WRITE(6,2041)S(I,5),S(I,1)
2041 FORMAT(T16,F4.1,' INCHES: SURFACE TREATMENT',T55,'CBR=',F
      >>>>
1990 GO TO 2045
      IF(I.GT.1) GO TO 8541

```



```

210      WRITE(6,8501)S(I,5)
      8501  FORMAT(T16,F4.1," INCHES OF ASPHALTIC PAVEMENT")
      GO TO 2045
      8541  WRITE(6,8502)S(I,5),S(I,1)
      8502  FORMAT(T16,F4.1," INCHES OF ASPHALTIC PAVEMENT",T55,"CBR
1 FTN 4.8+528      81/11/03, 09.30.59 PAGE 5
SUBROUTINE BASE      73/74      TS
>>>>      =" ,F4.1)
215      GO TO 2045
2000  IF(I.GT.1) GO TO 8542
      WRITE(6,8503)S(I,5)
      8503  FORMAT(T16,F4.1," INCHES OF BASE COURSE")
      GO TO 2045
220 8542  WRITE(6,8504)S(I,5),S(I,1)
      8504  FORMAT(T16,F4.1," INCHES OF BASE COURSE",T55,"CBR=" ,F4.1
>>>>      )
2010  IF(I.GT.1) GO TO 8543
      WRITE(6,8505)S(I,5)
      225 8505  FORMAT(T16,F4.1," INCHES OF SUBBASE MATERIAL")
      GO TO 2045
      8543  WRITE(6,8506)S(I,5),S(I,1)
      8506  FORMAT(T16,F4.1," INCHES OF SUBBASE MATERIAL",T55,"CBR="
>>>>      :F4.1)
230 2020  IF(I.GT.1) GO TO 8544
      WRITE(6,8507)S(I,5)
      8507  FORMAT(T16,F4.1," INCHES OF SELECT MATERIAL")

```



```

235      8544      60 TO 2045
      8508      WRITE(6,8508)S(I,5),S(I,1)
      F4,I,1)  FORMAT(T16,F4,1," INCHES OF SELECT MATERIAL",T55,"CBR=",
      >>>>
240      2045      DO 2055 J=1,L
      2050      WRITE(6,2050)
      2055      FORMAT(/)
      CONTINUE
      K=I+1
      IF(S(K,4).GT.0.0.OR.S(K,4).LT.0.0) GO TO 2075
      K=10
      2075      WRITE(6,2076)S(K,4)
      2076      FORMAT(T2,F4,1,T12,36("-"))
      245      2080      CONTINUE
      2085      WRITE(6,2086)S(10,1)
      2096      FORMAT(T22,"SUBGRADE",T55,"CBR=",F4,1,/,
      C60("X"))
      250      DO 2115 J=1,4
      DO 2110 I=1,10
      R(I,J)=0.0
      CONTINUE
      2110      CONTINUE
      2115      CONTINUE
      2119      WRITE(6,2120)
      255      2120      FORMAT("IF THE SUBGRADE IS COHESIVE(PI>5),TYPE 3",/,
      C"IF THE SUBGRADE IS NON-COHESIVE,TYPE 4.")
      READ(5,2130)K
      2130      FORMAT(I1)
      IF(K.EQ.3.OR.K.EQ.4) GO TO 2150

```



```

260      WRITE(6,2140)
2140     FORMAT('YOU MISUNDERSTOOD YOUR OPTIONS. TRY AGAIN.')
      GO TO 2119
2150     R(2,1)=S(2,4)+S(2,5)
      R(2,2)=100.
265     A1=70.7932+14.3841*I
      R1=-0.677999-0.136691*I
      A2=119.487+25.3872*I
      R(1,1)=S(1,4)+S(1,5)
      R(1,2)=100.
      IF(S(1,1).LT.20.) GO TO 2210
      GO TO 2230
      IF(K.EQ.4) GO TO 2225
      R(I,2)=90.
      GO TO 2230
      R(I,2)=95.
      CONTINUE
280     J=0
      DO 2260 ICC=1,4
      I=100-(5*ICC)
      J=J+1
      IDO=(A1+R1*I)+0.5
      R(J,3)=IDO
      IEE=(A2+R2*I)+0.5
      R(J,4)=IEE
      CONTINUE
      WRITE(6,2265)
      FORMAT(60(' '),/)
290 2265

```

6

PAGE

TS

73/74

1 FTN 4.8+528


```

C0X, "COMPACTION REQUIREMENTS (WRITTEN)", //
WRITE(6, 2280)
FORMAT("DEPTH", //
C"(INCHES)", // 7X, 7("=", ")", " TOP OF PAVEMENT ", 7("=", ")))
295 P=0,
DO 2365 I=2, 9
IF(R(I, 1), EQ, 0.) GO TO 2370
L=(R(I, 1)-P)/2, -1,
DO 2320 J=1, L
WRITE(6, 2315)
FORMAT(/)
CONTINUE
2325 WRITE(6, 2325)R(I, 2), S(I, 1)
FORMAT(T15, F5, 1, "% CBR=", // F4, 1)
ZOUCH=L
P=P+ZOUCH+1,
L=R(I, 1)-P-1,
DO 2350 J=1, L
WRITE(6, 2345)
FORMAT(/)
CONTINUE
2355 WRITE(6, 2355)R(I, 1)
FORMAT(T3, F4, 1, T10, 36("=", ")))
ZACH=L
P=P+ZACH+1,
CONTINUE
2365 WRITE(6, 2371)R(10, 2), S(10, 1)
FORMAT(T15, F5, 1, "% CBR=", // F4, 1, //
CT15, "TOP (TO DEPTH OF FILL)", // //
C60("=", ")", // T21, "FINAL DESIGN", // //
CT19, "FOR FILL SECTIONS")
C
C SUBROUTINE 2660
CALL FILLCUT(S, R, N, J1, K)

```



```

325      ZDD=J1
      J=21.-ZDD*0.2
      IF(R(10,2).GE.J1.OR.R(6,3).EQ.R(J,K)) GO TO 2430
      T=R(6,3)+4.
      WRITE(6,2420)S(10,1),J1
330 2420  FORMAT(T11,"4 INCHES,CBR=",F4.1," AT",F4.1,"% CE55")
      WRITE(6,2425)T
2425  FORMAT(T2,F4.1,T11,41(" "))
2430  WRITE(6,2431)S(10,1),R(10,2)
2431  FORMAT(T12,"TO DEPTH OF FILL,CBR=",F4.1," AT ",F4.1,"% C
>>>> E55"/
335      C," TDF",7X,F46(" "))
      T=R(6,4)-S(10,2)
      IF(T.LT.10.) GO TO 2465
      T=S(10,2)+5.
      WRITE(6,2455)T
340 2455  FORMAT(T11,"6 INCHES,SCARIFY AND COMPACT IN PLACE AT",F5
>>>> ,1,"% CE55
      C"/,T11,47(" "))
2465  WRITE(6,2466)S(10,1),S(10,2)
2466  FORMAT(T11,"CBR=",F4.1," IN-PLACE DENSITY=",F4.1,"% CE55
>>>> "/,
345      C60(" *"),/,
      CT21,"FINAL DESIGN FOR CUT SECTIONS")
      C
      C
      SUBROUTINE 2660
      CALL  FILLCUT(S,R,N,J1,K)
      L=6
350      DO 2545 J=1,5

```



```

J1=100-(J-1)*5
IF(R(J,K),LE,R(L,3)) GO TO 2545
IF(J1,LE,S(10,2)) GO TO 2550
L=L+1
355 ZGG=R(J,K)-R(L-1,3)
IF(ZGG,LT,4.) GO TO 2535
R(L,3)=R(J,K)
GO TO 2540
2535 R(L,3)=R(L-1,3)+4.
360 R(L,4)=J1
2545 CONTINUE
2550 ZHH=R(L,3)-R(L-1,3)
IF(ZHH,LE,6.) GO TO 2560
R(L-1,3)=R(L,3)-6.
365 2560 DO 2610 J=7,L
N=(R(J,3)-P)/2.-1.
C
C SUBROUTINE 2635
CALL SPACER(N,P)
370 IF(J,EO,L) GO TO 2590
WRITE(6,2590)R(J,4)
2580 FORMAT(T11,"SUBGRADE REMOVED AND REPLACED AT ",F5.1,"% C
E55")
>>>>
2590 GO TO 2595
375 WRITE(6,2591)R(J,4)
E55")
>>>>
2595 N=R(J,3)-P-1.
1 FTN 4,8+528 81/11/03, 09.30.59 PAGE 8
SUBROUTINE BASE 73/74 TS
C SUBROUTINE 2635

```



```

380      CALL SPACER(N,P)
      WRITE(6,2605)R(J,3)
2605     FORMAT(T2,F4.1,T11,36("-"))
2610     CONTINUE
      WRITE(6,2615)S(10,1),S(10,2)
2615     FORMAT(T11,"CBR=",F4.1,5X,"IN PLACE DENSITY=",F5.1,"5 CE
>>>> 55"/,
395     C60("X"),/,
      C"THIS COMPLETES THE SOILS DESIGN")
      RETURN
      END

1 45000B CM STORAGE USED      6.795 SECONDS
1 FTN 4.84529      91/11/03. 09.30.59 PAGE 1
SUBROUTINE SPACER 73/74 TS

C
C
C
5     SUBROUTINE 2635
      SUBROUTINE SPACER(N,P)
      DO 2645 IQ=1,N
      WRITE(6,2460)
2460     FORMAT(/)
2645     CONTINUE
      ZII=N
      P=P+ZII+1.
      RETURN
      END
10

1 41000B CM STORAGE USED      .052 SECONDS
1 FTN 4.94529      91/11/03. 09.30.59 PAGE 1
SUBROUTINE FILLCUT 73/74 TS

C
C     SUBROUTINE 2660

```



```

5      SUBROUTINE FILLCUT(S,R,N,J1,K)
        DIMENSION S(10,6),R(20,4)
        J1=80
        WRITE(6,2670)
        FORMAT(" DEPTH",/,
2670  C"(INCHES)",3X,14("="), " TOP OF PAVEMENT",1X,14("="))
        IF(S(1,5).GT.1.) GO TO 2710
        IF(S(1,5).EQ.1.) GO TO 2695
        WRITE(6,2685)
        FORMAT(T11,"SURFACE TREATMENT REQUIRED")
2685  GO TO 2700
        WRITE(6,2696)
        FORMAT(T11,"MULTIPLE SURFACE TREATMENT REQUIRED")
15 2696 P=1.
        GO TO 2725
        N=(S(1,5))/2.-1.)
        C
20 C      SUBROUTINE 2625
        CALL SPACER(N,P)
        WRITE(6,2716)S(1,5)
        FORMAT(T11,F4.1," INCHES OF ASPHALTIC PAVEMENT")
2716  N=5(1,5)-P
        C
25 C      SUBROUTINE 2635
        CALL SPACER(N,P)
        WRITE(6,2720)S(1,5)
        FORMAT(T2,F4.1,T11,36(" "))
        DO 2725 I=2,8
30 2720 IF(S(I,1).EQ.0,0) GO TO 2960
        IF(S(I,1).LT.20,0) GO TO 2785
        N=(R(I,1)-P)/2.-1.

```



```

C
35 C SUBROUTINE 2635
CALL SPACER(N,P)
WRITE(6,2750)S(I,5),S(I,1)
2750 FORMAT(T11,F4.1," INCHES,CER=",F5.1," AT 100% CE55")
N=(R(I,1)-P-1.)
40 C
C SUBROUTINE 2635
CALL SPACER(N,P)
WRITE(6,2765)R(I,1)
2765 FORMAT(T2,F4.1,T11,36(" "))
R(6,3)=R(I,1)
R(6,4)=100.0
GO TO 2960
2785 L=6
DO 2830 J=1,5
J1=100-(J-1)*5
IF(R(J,K),LT,R(L,3)) GO TO 2830
IF(R(I,1),LT,R(J,K)) GO TO 2835
IF(R(I,2),LE,J1) GO TO 2830
L=L+1
R(L,K)=R(J,3)
R(L,4)=J1
2830 CONTINUE
1 FTN 4,8+528 81/11/03, 09.30.59 PAGE 2
SUBROUTINE FILLCUT 73/74 TS
2835 L=L+1
R(L,3)=R(I,1)
IF(R(I,2),LT,J1) GO TO 2855
R(L,4)=R(I,2)
GO TO 2865
60

```



```

2855 R(L,4)=JI
2865 IKK=L-1
65 DO 2885 J=6,IKK
ZFINK=R(J+1,3)-R(J,3)
IF(ZFINK,GE,4.) GO TO 2885
LFINK=L-1
70 IF(J.EQ,LFINK) GO TO 2895
R(J+1,3)=R(J,3)+4.
CONTINUE
GO TO 2905
2895 R(J,3)=R(L,3)
L=L-1
75 2905 DO 2945 J=7,L
K=(R(J,3)-P)/2.-1.
C
C SUBROUTINE 2635
CALL SPACER(N,P)
80 T=R(J,3)-R(J-1,3)
WRITE(6,2925)T,S(I,1),R(J,4)
2925 FORMAT(T11,F4.1," INCHES,CER=",F4.1," AT",F5.1,"% CESS")
N=R(J,3)-P-1.
C
C SUBROUTINE 2635
CALL SPACER(N,P)
85 WRITE(6,2940)R(J,3)
2940 FORMAT(T2,F4.1,T11,36(" "))
CONTINUE
90 R(6,3)=R(L,3)
R(6,4)=R(L,4)
CONTINUE
RETURN
END
2960

```

41008 CM STORAGE USED .814 SECONDS


```

30      READ(5,4075)E(I,2)
      FORMAT(F7.3)
      WRITE(6,4080)
4080    FORMAT("WHAT IS CORRESPONDING ELEVATION CENTERLINE?")
      READ(5,4085)E(I,3)
      FORMAT(F7.3)
      WRITE(6,4090)
4090    FORMAT("WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?")
      READ(5,4095)E(I,4)
      FORMAT(F7.3)
      CONTINUE
40     DO 4115 I=1,M
      E(I,5)=0.0
      J=I+1
      E(J,5)=(E(J,3)-E(I,3))/(E(J,1)-E(I,1))
45     CONTINUE
      WRITE(6,4120)
4120    FORMAT(/,T2,"STA NO",T15,"LS ELEV",T30,"CL ELEV",T45,"RS
      ELEV",T60
      C,"SLOPE",/)
      DO 4140 I=1,M
      WRITE(6,4135)E(I,1),E(I,2),E(I,3),E(I,4),E(I,5)
4135    FORMAT(T2,F7.3,T15,F7.3,T30,F7.3,T45,F7.3,T60,F7.3)
4140    CONTINUE
1 FTN 4.8+528      81/11/03. 09.30.59 PAGE      2
SUBROUTINE ELEV      73/74      TS
      WRITE(6,4150)
4150    FORMAT(/,"YOU MUST REVIEW THE SLOPE AND DETERMINE THE ",
      /)
      C="DESIRED GRADE OR GRADES."/)
55

```



```

M1=M-1
DO 4567 I=1,M1
  G(I)=0.0
CONTINUE
4567 DO 4175 I=1,M1
      J=I+1
      WRITE(6,4165)E(I,1),E(J,1)
      FORMAT('WHAT IS THE GRADE FROM ',F6.2,' TO ',F6.2)
65 4170 READ(5,4170)G(I)
      FORMAT(F7.3)
      CONTINUE
      C
      C
      C
70 4180 WRITE(6,4180)
      FORMAT(/,'WHAT IS THE FINAL GRADE CENTERLINE ELEVATION',
      >>>> /,
      C"OF THE FIRST STATION ?")
      READ(5,4185)E(I,7)
      FORMAT(F7.3)
      DO 4214 I=1,M1
        J=I+1
        E(J,7)=((E(J,1)-E(I,1))*G(I))+E(I,7)
        CONTINUE
        I=1
80 4220 JK=I+1
      IF(G(I),EQ,6(JK)) GO TO 4275
      WRITE(6,4221)E(JK,1)
      FORMAT(/,'A VERTICAL CURVE IS NECESSARY AT STA ',F6.2)
      WRITE(6,4225)
95 4225 FORMAT('HOW MANY STATIONS DO YOU WANT TO INCLUDE IN THE

```



```

115      A=15.5*H(J,8)
        C=A/(H(J,5)/2.)
        Q=11.5*H(J,8)
        R=Q/(H(J,5)/2.)
        IF(E(I,6).EQ.2.) GO TO 4525
4345      HCUCH=H(J,6)+(H(J,5)/200.)
120      IF(E(I,1).GT.HCUCH) GO TO 4410
        D=(E(I,1)-H(J,6))*C
        N=(E(I,1)-H(J,6))*R
        E(I,13)=E(I,7)+D
        E(I,12)=E(I,7)+W
125      E(I,14)=E(I,13)-4.5*0.333
        E(I,15)=E(I,14)+8.1*0.275
        E(I,11)=E(I,7)-11.5*0.275
        E(I,10)=E(I,11)-4.0*0.63
        E(I,9)=E(I,10)-4.5*0.333
130      E(I,8)=E(I,9)+8.1*0.275
        I=I+1
        GO TO 4345
4410      HIA=H(J,7)-(H(J,5)/200.)
        IF(E(I,1).GT.HIA) GO TO 4455
135      E(I,13)=E(I,7)+A
        E(I,12)=E(I,7)+R
        E(I,14)=E(I,13)-4.5*0.333
        E(I,15)=E(I,14)+8.1*0.275
        E(I,10)=E(I,11)-4.0*0.63
        E(I,9)=E(I,10)-4.5*0.333
        E(I,8)=E(I,9)+8.1*0.275
        I=I+1
        GO TO 4410
4455      P=(H(J,7)-E(I,1))*C
145      T=(H(J,7)-E(I,1))*R
        E(I,13)=E(I,7)+P
        E(I,12)=E(I,7)+T
        E(I,14)=E(I,13)-4.5*0.333
        E(I,15)=E(I,14)+8.1*0.275
        E(I,11)=E(I,7)-11.5*0.42
150

```



```

E(I,10)=E(I,11)-4.0*0.063
E(I,9)=E(I,10)-4.5*0.333
E(I,8)=E(I,9)+8.1*0.275
I=I+1
155 IF(E(I,1),LT,H(J,7)) GO TO 4455
J=J+1
GO TO 4315
4525 HTR=H(J,6)+(H(J,5)/200.)
IF(E(I,1),GT,HIB) GO TO 4595
D=(E(I,1)-H(J,6))*C
W=(E(I,1)-H(I,6))*R
E(I,10)=E(I,7)+D
E(I,11)=E(I,7)+W
160 81/11/03, 09.30.59 PAGE 4
SURROUTINE ELEV 73/74 TS
E(I,9)=E(I,10)-4.5*0.333
E(I,8)=E(I,9)+8.1*0.275
E(I,12)=E(I,7)-11.5*0.042
E(I,13)=E(I,12)-4.0*0.063
E(I,14)=E(I,13)-4.5*0.333
E(I,15)=E(I,14)+8.1*0.275
I=I+1
GO TO 4525
4595 HIC=H(J,7)-(H(J,5)/200.)
IF(E(I,1),GT,HIC) GO TO 4650
E(I,10)=E(I,7)+A
E(I,11)=E(I,7)+R
E(I,9)=E(I,10)-4.5*0.333
E(I,8)=E(I,9)+8.1*0.275
E(I,12)=E(I,7)-11.5*0.042
E(I,13)=E(I,12)-4.0*0.063
E(I,14)=E(I,13)-4.5*0.333
E(I,15)=E(I,14)+8.1*0.275
170
175
180

```



```

I=I+1
GO TO 4595
4650 P=(H(J,7))-E(I,1))*C
185 T=(H(J,7))-E(I,1))*R
E(I,10)=E(I,7)+F
E(I,11)=E(I,7)+T
E(I,9)=E(I,10)-4.5*0.333
E(I,8)=E(I,9)+8.1*0.275
190 E(I,12)=E(I,7)-11.5*0.042
E(I,13)=E(I,12)-4.0*0.063
E(I,14)=E(I,13)-4.5*0.333
E(I,15)=E(I,14)+8.1*0.275
I=I+1
195 4710 TF(E(I,1),LT,H(J,7)) GO TO 4650
J=J+1
GO TO 4315
4725 E(I,11)=E(I,7)-11.5*0.042
E(I,10)=E(I,11)-4.0*0.063
200 E(I,9)=E(I,10)-4.5*0.063
E(I,8)=E(I,9)+8.1*0.275
E(I,12)=E(I,7)-11.5*0.042
E(I,13)=E(I,12)-4.0*0.063
E(I,14)=E(I,13)-4.5*0.333
E(I,15)=E(I,14)+8.1*0.275
I=I+1
MIZ=M+1
IF(I,EQ,MIZ) GO TO 4780
GO TO 4315
210 4780 WRITE(6,4781)
4781 FORMAT(I2,"STA",T22,"BACKL",T42,"DITL",T62,"SHL",T82,"FU
>>>> MTL",T102,
C'CL'://)

```



```

DO 4800 I=1,M
WRITE(6,4795)E(I,1),E(I,8),E(I,9),E(I,10),E(I,11),E(I,7)
FORMAT(T2,F7.3,T22,F7.3,T42,F7.3,T62,F7.3,T82,F7.3,T102,
>>>> F7.3)
4800 CONTINUE
WRITE(6,4815)
4815 FORMAT(/,T2,"STA",T22,"FUMTR",T42,"SHR",T62,"DIIR",T82,
1 FTN 4.8+528 81/11/03, 09.30.59 PAGE 5
SUBROUTINE ELEV 73/74 TS

>>>> "BACKR",/)
DO 4835 I=1,M
WRITE(6,4830)E(I,1),E(I,12),E(I,13),E(I,14),E(I,15)
4830 FORMAT(T2,F7.3,T22,F7.3,T42,F7.3,T62,F7.3,T82,F7.3)
4835 CONTINUE
CALL XSECT(E,M)
RETURN
END
225

43000B CM STORAGE USED 3.828 SECONDS
1 FTN 4.8+528 81/11/03, 09.30.59 PAGE 1
SUBROUTINE XSECT 73/74 TS

C SUBROUTINE XSECT
C
C THIS SUBROUTINE COMPUTES THE CROSS SECTIONAL AREA OF
5 C EACH STATION, AND OUTPUTS THE VOLUME AND CUMULATIVE
C VOLUME BETWEEN STATIONS. IT ALSO COORDINATES THE
C OUTPUTS OF THE INSECT AND MASBAL SUBROUTINES.
C SUBROUTINE XSECT(E,M)

```



```

C
C THIS SUBROUTINE COMPUTES THE CROSS SECTIONAL AREA OF
C EACH STATION, AND OUTPUTS THE VOLUME AND CUMULATIVE
5 C VOLUME BETWEEN STATIONS. IT ALSO COORDINATES THE
C OUTPUTS OF THE INSECT AND MASBAL SUBROUTINES.
C
SUBROUTINE XSECT(E,M)
10 DIMENSION E(50,15)
DIMENSION INGR(50),INGL(50)
DIMENSION XINSR(50,2),XINSL(50,2)
DIMENSION YS(12),XS(12)
DIMENSION XSA(50)
15 DIMENSION VOL(50),CVOL(50)
DIMENSION X(52),Y(52)
DIMENSION YMAX(50)
DIMENSION RUE(512)
DATA YS(1),YS(2),XS(3),XS(4),XS(5),XS(6),XS(7),XS(8),XS(
20 C9),XS(10),XS(12)/0.0,28.1,20.0,15.5,11.5,0.0,-11.5,-15.5
C,-20.0,-28.1,0.0/
CALL INSECT(E,M,XINSR,XINSL,INGR,INGL)
DO 5000 J=1,M
YS(2)=XINSL(J,1)-E(J,8)
YS(3)=XINSL(J,1)-E(J,9)
25 YS(4)=XINSL(J,1)-E(J,10)
YS(5)=XINSL(J,1)-E(J,11)
YS(6)=XINSL(J,1)-E(J,7)
YS(7)=XINSL(J,1)-E(J,12)
YS(8)=XINSL(J,1)-E(J,13)
YS(9)=XINSL(J,1)-E(J,14)
30 YS(10)=XINSL(J,1)-E(J,15)
YS(11)=XINSL(J,1)-XINSR(J,1)
YS(12)=XINSL(J,1)-E(J,3)
YS(1)=XINSL(J,2)

```



```

XS(11)=XTNR(J,2)*(-1.0)
XSA(J)=(YS(12)*XS(1)-XS(11))+(YS(11))*XS(2)-XS(12))
DO 5001 I=2,11
  NXSA=I+1
  MXSA=I-1
  XSA(J)=XSA(J)+(YS(I))*XS(NXSA)-XS(MXSA))
CONTINUE
5001 XSEXT=(XSA(J)/2.0)
IF(XSEXT.GE.0.0) GO TO 5500
XSA(J)=XSEXT*(-1.0)
GO TO 5501
5500 XSA(J)=XSEXT
IF(E(J,7).GT.E(J,3)) GO TO 5000
XSA(J)=XSA(J)*(-1.0)
CONTINUE
VOL(1)=0.0
DO 5002 J=2,M
VOL(J)=(XSA(J-1)+XSA(J))*E(J,1)-E(J-1,1))*100.0/54.0
CONTINUE
5002 CVOL(1)=0.0
SUM=0.0
DO 5003 J=2,M
  81/11/03. 09.30.59 PAGE 2
  SUBROUTINE XSECT 73/74 TS
CVOL(J)=VOL(J)+SUM
SUM=CVOL(J)
CONTINUE
5003 WRITE(6,5004)
5004 FORMAT(/,/,51('#'))
5005 WRITE(6,5005)
5005 FORMAT("SUBROUTINE XSECT COMPUTES CROSS SECTIONAL AREAS"
/ / /
/ / /
65 C"FOR EACH STA, VOLUME BETWEEN STAS, AND CUMULATIVE",/
C"VOLUME, FILL AREAS ARE GIVEN A + SIGN AND CUT AREAS",/
C"ARE INDICATED BY A - SIGN.",/ / /
WRITE(6,5006)
5006 FORMAT(/,/,21('#')," VOLUMES ",21('#'),/ / /
WRITE(6,5007)
70

```



```

>>>> 5007 FORMAT(2X,"STA NO",2X,"CROSS SECTION",2X,"VOLUME BETWEEN
      ",2X,"CUMU
      CLATIVE")
      WRITE(6,5008)
5008  FORMAT(12X,"AREAS(SF)",5X,"STATIONS(CY)",3X,"VOLUME(CY)"
      ,//)
75  DO 5009 J=1,M
      WRITE(6,5010)VOL(J),CVOL(J)
5010  FORMAT(27X,F10.2,4X,F10.2)
      WRITE(6,5011)E(J,1),XSA(J)
5011  FORMAT(2X,F6.2,2X,F10.2)
80 5009  CONTINUE
      WRITE(6,5012)
5012  FORMAT(//,51('#='))
      CALL MMSBAL(E,M,CVOL)
      WRITE(6,5013)
5013  FORMAT(//,61('*='))
      RETURN
      END

41000B CM STORAGE USED          .799 SECONDS
1 FTN 4.8+528          81/11/03. 09.30.59 PAGE 1
SUBROUTINE INSECT 73/74 TS

C SUBROUTINE INSECT
C
C THIS SUBROUTINE APPROXIMATES THE INTERSECTION
5 C POINT OF THE ROAD SIDESLOPES WITH THE
C EXISTING GROUND. INPUT CONSISTS OF THE
C SIDESLOPE GRADE AT EACH STATION.
C
SUBROUTINE INSECT(E,M,XINSR,XINSL,INGR,INGL)
10 DIMENSION E(50,15)

```



```

DIMENSION XINSR(50,2),XINSL(50,2)
DIMENSION INGR(50),INGL(50)
DO 6010 I=1,M
  XINSR(I,1)=0.0
  XINSR(I,2)=0.0
  XINSL(I,1)=0.0
  XINSL(I,2)=0.0
6010 CONTINUE
  WRITE(6,6011)
20 6011 FORMAT(//,'61(**)',//,56('='),//,
  C'SUBROUTINE INSECT COMPUTES THE INTERSECTION OF THE',//,
  C'SIDESLOPES WITH THE EXISTING GROUND AT EACH STA.',//,
  C'INPUT A SLOPE GRADE IN RESPONSE TO THE PROGRAMS QUESTIO
  >>>> NS.',//
25 C'USE ANY WHOLE NUMBER UP TO TWO DIGITS.',//
  C'I.E. USE 2 FOR 1:2 OR 4 FOR 1:4',//)
  DO 6012 J=1,M
6017 WRITE(6,6013)E(J,1)
6013 FORMAT('WHAT IS THE RIGHT SLOPE GRADE AT STA. ',F6.2,'?')
  >>>>
30 6014 READ(5,6014)INGR(J)
  FORMAT(I2)
6015 WRITE(6,6015)
  FORMAT(/)
  TCA=INGR(J)
  IF(E(J,15).EQ.E(J,4)) GO TO 6025
  IF(E(J,15).LT.E(J,4)) GO TO 6021
  TCB=(E(J,3)-E(J,4))/28.1
  IF(TCB.NE.TCA) GO TO 6019
  WRITE(6,6016)
6016 FORMAT('THE SLOPE GRADE CHOSEN IS PARRALLEL WITH',//,
  C'THE EXISTING GROUND. CHOOSE ANOTHER SLOPE GRADE.',//)
40

```



```

1 FTN 4.8+528          21/11/03, 09.30.59 PAGE    2
SUBROUTINE INSECT    73/74    TS

45      GO TO 6017
6018     I=1
6019     XAI=I
        TCD=E(J,4)-(TCB**XAI)
        TCE=28.1+(TCA**XAI)
        TCF=E(J,15)-XAI
        IF(TCF.LE.TCD) GO TO 6020
        I=I+1
        GO TO 6019
50 6020     XINSR(J,1)=TCD
        XINSR(J,2)=TCE
        GO TO 6012
6021     TCC=(E(J,4)-E(J,3))/28.1
        IF(TCC.NE.TCA) GO TO 6023
        WRITE(6,6022)
55      21/11/03, 09.30.59 PAGE    2
SUBROUTINE INSECT    73/74    TS

6022     FORMAT('THE SLOPE GRADE CHOSEN IS PARRALLEL WITH',/,
C'THE EXISTING GROUND. CHOOSE ANOTHER SLOPE GRADE',/)
        GO TO 6017
6023     I=1
6024     XBI=I
        TCG=E(J,4)+(TCC**XBI)
        TCH=28.1+(TCA**XBI)
        YCI=E(J,15)+XBI
        IF(TCI.GE.TCG) GO TO 6026
        I=I+1
        GO TO 6024
6026     XINSR(J,1)=TCI
        XINSR(J,2)=TCH
        GO TO 6012
70 6025     XINSR(J,1)=E(J,15)
        XINSR(J,2)=28.1
6012     CONTINUE
        DO 6030     J=1,M
6031     WRITE(6,6032)E(J,1)

```



```

75 6032 FORMAT("WHAT IS THE LEFT SLOPE GRADE AT STA. ",F6.2,"?")
      6034 READ(5,6034)INGL(J)
      6035 FORMAT(I2)
      6036 WRITE(6,6035)
      6037 FORMAT(/)
      6038 TCAA=INGL(J)
      6039 IF(E(J,8),EQ,E(J,2)) GO TO 6045
      6040 IF(E(J,8),LT,E(J,4)) GO TO 6040
      6041 TCJ=(E(J,3)-E(J,2))/28.1
      6042 IF(TCJ,NE,TCAA) GO TO 6037
      6043 WRITE(6,6036)
      6044 FORMAT("THE SLOPE GRADE CHOSEN IS PARRALLEL WITH",/,
      6045 C"THE EXISTING GROUND, CHOOSE ANOTHER SLOPE GRADE.",/,)
      6046 GO TO 6031
      6047 I=I
      6048 XCI=I
      6049 TCL=E(J,2)-(TCJ*XCI)
      6050 TCM=28.1+(TCAA*XCI)
      6051 TCN=E(J,8)-XCI
      6052 IF(TCN,LE,TCL) GO TO 6039
      6053 I=I+1
      6054 GO TO 6038
      6055 XINSL(J,1)=TCL
      6056 XINSL(J,2)=TCM
      6057 GO TO 6030
      6058 TCN=(E(J,2)-E(J,3))/28.1
      6059 IF(TCN,NE,TCAA) GO TO 6042
      6060 WRITE(6,6041)
      6061 FORMAT("THE SLOPE GRADE CHOSEN IS PARRALLEL WITH",/,
      6062 C"THE EXISTING GROUND, CHOSE ANOTHER SLOPE GRADE.",/,)
      6063 GO TO 6031

```



```

6042 I=1
6043 XDI=I
TCQ=E(J,2)+(TCK*XDI)
TCP=29.1+(TCAA*XDI)
TCQ-E(J,8)+XDI
110 IF(TCQ,GE,TCO) GO TO 6044
I=I+1
1 FTN 4.8+528 81/11/03. 09.30.59 PAGE 3
SUBROUTINE INSECT 73/74 TS
GO TO 6043
6044 XINSL(J,1)=TCQ
XINSL(J,2)=TCP
GO TO 6030
6045 XINSL(J,1)=E(J,8)
XINSL(J,2)=28.1
6030 CONTINUE
WRITE(6,6050)
6050 FORMAT(/,18('='),' INTERSECTION POINTS ',17('='),/)
WRITE(6,6051)
6051 FORMAT(2X,'STA NO',4X,' LEFT HAND SIDESLOPE',4X,'RIGHT H
AND SIDESL
COPE')
125 WRITE(6,6052)
FORMAT(13X,'GRADE ELEV DIST',6X,'GRADE ELEV DIST',
/>)
DO 6053 J=1,M
WRITE(6,6054)E(J,1),INCL(J),XINSL(J,1),XINSL(J,2),INGR(J
),XINSR(J,
C1),XINSR(J,2)
130 6054 FORMAT(2X,F6.2,2,6X,I2,2X,F6.2,1X,F6.2,7X,I2,2X,F6.2,1X,F6

```



```

>>>>      .2)
6053      CONTINUE
          WRITE(6,6060)
6060      FORMAT(//,56('#=#'))
          RETURN
135      END

          42000B CM STORAGE USED      1.575 SECONDS
1 FTN 4.8+528      81/11/03. 09.30.59 PAGE 1
SUBROUTINE MABAL 73/74 TS

C
C      SUBROUTINE MABAL
C
C      THIS SUBROUTINE COMPUTES THE MAXIMUM
5 C      AND MINIMUM VOLUMES AND IDENTIFIES THEIR
C      STATION LOCATION. IT ALSO PLOTS A
C      MASS DIAGRAM ON THE VERSATED PLOTTER.
C

10      SUBROUTINE MABAL(E,M,CVOL)
          DIMENSION E(50,15),CVOL(50)
          DIMENSION X(52),Y(52),YMAX(50)
          DIMENSION BUF(512)
          DO 8800 J=1,M
            X(J)=E(J,1)
            Y(J)=CVOL(J)
          CONTINUE
15      8800      DO 8900 J=1,M
                Y(J)=Y(J)+(1000000.0)
          CONTINUE
                I=1
20

```



```

8810 AMAX=Y(I)
      K=1
      J=2
8808 IF(Y(J).LT.AMAX) GO TO 8805
      AMAX=Y(J)
      K=J
8805 IF(J.EQ.M) GO TO 8806
      GO TO 8807
8806 YMAX(I)=AMAX
      Y(K)=0.0
      GO TO 8809
8907 J=J+1
      GO TO 8808
8909 IF(I.EQ.M) GO TO 8811
      I=I+1
      GO TO 8810
8911 DO 8901 J=1,M
      Y(J)=YMAX(J)-(1000000.0)
8901 CONTINUE
      WRITE(6,8700)
8700 FORMAT(//,60(=))
      WRITE(6,8701)
8701 FORMAT("SUBROUTINE MASBAL FINDS MAXIMUM AND MINIMUM",/,
C" CUMULATIVE VOLUMES, AND IDENTIFIES THEIR STATION",/,
C" LOCATION POINTS. IT ALSO PLOTS A MASS DIAGRAM ON",/,
C" THE VERSATEC PLOTTER, NO INPUT IS NECESSARY.",//)
      J=1
      YMAX(1)=YMAX(1)-(1000000.0)
50 8851 IF(YMAX(1).EQ.Y(J)) GO TO 8850
      J=J+1
      GO TO 8851
8850 WRITE(6,8852)Y(J),X(J)
8852 FORMAT("THE MAXIMUM CUMULATIVE VOLUME IS",F10.2," CUBIC
      YARDS",/)

```

```

***

```



```

55      C"THIS OCCURS AT STATION ",F6.2,/)
      J=1
1 FTN 4.84528      81/11/03. 09.30.59 PAGE      2
SUBROUTINE MASBAL 73/74  TS
      YMAX(M)=YMAX(M)-(1000000.0)
8854  IF(YMAX(M).EQ.Y(J)) GO TO 8853
      J=J+1
60      GO TO 8854
8853  WRITE(6,8855)Y(J),X(J)
8855  FORMAT("THE MINIMUM CUMULATIVE VOLUME IS",F10.2," CUBIC
>>>>  YARDS",/)
      C"THIS OCCURS AT STATION ",F6.2,/)
65      DO 8799 I=1,M
      Y(J)=Y(J)/1000.
      CONTINUE
8799  C
      OPEN THE PLOTFILE AND EXPAND PAGE LIMITS
      CALL PLOTS (BUF,512,9,00)
      CALL PLOTMX (55.)
70  C      SCALE THE INPUT TO THE PAPER SIZE (6" X 50")
      CALL SCALE (X,50.,50,1)
      CALL SCALE (Y,6.,50,1)
      C      PLOT AND NAME THE AXIS
      CALL AXIS (1.,1.,"1000 CUBIC YARDS",16,6.,90.,Y(51),Y(52)
>>>> )
75  C      CALL AXIS (1.,1.,"STATIONS",-9.50,0.,X(51),X(52))
      PLOT ARRAY
      CALL LINE (X,Y,50,1,-1,3)
      C      CLOSE FILE
      CALL PLOT (0.,0.,999)
      WRITE(6,9950)
80  C      FORMAT(/,"GO TO THE PLOTTER BIN AND REMOVE OUTPUT (MASS
>>>>  DIAGRAM)"
      C,/)

```



```

CALL SCALE (X,50,50,1)
CALL SCALE (Y,6,50,1)
C
PLOT AND NAME THE AXIS
CALL AXIS (1,1,1, "1000 CUBIC YARDS",16.6,90.,Y(51),Y(52)
>>>> )
75
CALL AXIS (1,1,1, "STATIONS",-9.50,0.,X(51),X(52))
C
PLOT ARRAY
CALL LINE (X,Y,50,1,-1,3)
C
CLOSE FILE
CALL PLOT (0,0,999)
80
WRITE(6,9950)
9950 FORMAT(/, "GO TO THE PLOTTER BIN AND REMOVE OUTPUT (MASS
>>>> "DIAGRAM)"
C://)
WRITE(6,9951)
9951 FORMAT(/,60(" "))
95
RETURN
END

```

```

410008 CM STORAGE USED          .684 SECONDS
15.793 CP SECONDS COMPILATION TIME

```


APPENDIX F

SAMPLE OUTPUT


```

/OLD*FIXD
/ATTACH*CALCOMP/UN=LIBRARY
/LIBRARY(CALCOMP)
/LIBRARY(CALCOMP)
/FIN*1=FIXD*L=0*ITS
15.741 CP SECONDS COMPILATION TIME
/LGO.

```

***** ROAD DESIGN *****

THIS PROGRAM IS A SERIES OF SUBROUTINES DESIGNED TO ASSIST THE ENGINEER IN A MILITARY ROAD DESIGN. THE OPERATION OF THE PROGRAM IS INTERACTIVE, AND REQUIRES THE USE OF THE USERS MANUAL.

HORIZONTAL ALIGNMENT

ENTER NO. OF CURVES THIS SECTION
? 1

ENTER STA NO OF TANGENT INTERSECTION OF CURVE 1
? 133.0

ENTER ANGLE OF INTERSECTION OF CURVE IN DEGREES
? 26.0

ENTER CHORD LENGTH IN FEET
? 600.0

STA NO.	KAD	L OF C	U OF L	TAN LEN	PC STA	PI STA
132.947	1333.62	600.181	7.46/05	507.891	129.921	139.052

SUPERELEVATION SUPERL

WHAT IS DESIGN VELOCITY OF CURVE	1	STA NO	RAD	L OF C	SUPERL	LAN LEN	PC STA	PT STA
? 60		132.947	1333.62	605.181	.04996069	307.891	129.921	139.052

WHAT DESIGN INDEX IS TO BE USED?

? 6

WHAT IS THE IN-PLACE (%CESS) OF THE SUBGRADE?

? 70.0

WHAT IS THE CBR OF THE SUBGRADE AT 90% TO 95% DEPTH?

? 5.0

HOW MANY TYPES OF BASE, SUBBASE, AND SELECT MATERIALS

DO YOU WISH TO CONSIDER? (DO NOT INCLUDE SUBGRADE OR PAVT)

? 3

WHAT IS THE CBR VALUE OF THE BASE COURSE?

? 82.0

WHAT IS THE NEXT HIGHEST CBR?

? 56.0

WHAT IS THE NEXT HIGHEST CBR?

? 43.0

WHAT CLASSIFICATION IS TO BE USED?

A=1, B=2, C=3, D=4, E=5, F=6

? 3

11.0 INCHES OF SELECT MATERIAL

CBR=43.0

22.0

 SUBGRADE CBR= 5.0

 IF THE SUBGRADE IS COHESIVE (PI>5), TYPE 5
 IF THE SUBGRADE IS NON-COHESIVE, TYPE 4.
 4

 COMPACTION REQUIREMENTS (WRITTEN)

DEPTH (INCHES)

===== 10P OF PAVEMENT =====

100.0% CBR=82.0

7.0

100.0% CBR=56.0

11.0

100.0% CBR=43.0

22.0 -----
95.0% CBR= 5.0
TOP (TO DEPTH OF FILL)

FINAL DESIGN
FOR FILL SECTIONS
DEPTH (INCHES) ===== TOP OF PAVEMENT =====
3.0 INCHES OF ASPHALTIC PAVEMENT

3.0 -----
4.0 INCHES, CBR= 82.0 AT 100% LESS

7.0 -----
4.0 INCHES, CBR= 56.0 AT 100% LESS

11.0 -----

11.0 INCHES,CRK= 43.0 AT 100% CESS

22.0 -----
TO DEPTH OF FILL,CRK= 5.0 AT 95.0% CESS
IDF -----
6 INCHES,SCARIFY AND COMPACT IN PLACE AT 75.0% CESS

CRK= 5.0 IN-PLACE DENSITY=70.0% CESS

FINAL DESIGN FOR CUT SECTIONS
DEPTH
(INCHES) ===== 10' OF PAVEMENT =====

3.0 INCHES OF ASPHALTIC PAVEMENT

3.0

4.0 INCHES,CRK= 82.0 AT 100% CESS

7.0

4.0 INCHES,CRK= 56.0 AT 100% CESS

11.0

11.0 INCHES,CRK= 43.0 AT 100% CESS

22.0

SURGRADE REMOVED AND REPLACED AT 100.0% CESS

28.0

SURGRADE REMOVED AND REPLACED AT 95.0% CESS

40.0

SURGRADE REMOVED AND REPLACED AT 90.0% LESS

60.0 -----

SCARIFY AND COMPACT IN-PLACE AT 85.0% LESS

66.0 -----
CRK= 5.0 IN PLACE DENSITY= 70.05 LESS
***** THIS COMPLETES THE SOILS DESIGN *****

THIS SUBROUTINE WILL CALCULATE THE CENTERLINE ELEVATION OF THE FINISHED GRADE. YOU MUST SUPPLY THE ACTUAL ELEVATIONS OF THE LEFT SIDE DITCH BACKSLOPE, CENTERLINE AND THE RIGHT SIDE DITCH BACKSLOPE. THE TERMINAL POINT FOR BACKSLOPES IS 28 FEET LEFT OR RIGHT OF THE CENTERLINE. ENSURE THAT THE PC, PT, AND MIDPOINTS OF ANY CURVES IN THIS SECTION ARE INCLUDED.

HOW MANY STATIONS IN THIS SECTION? MAX=50

? 16

WHAT IS THE NEXT STATION NO.?

? 127.0

IS THIS STATION A CURVE PC?

IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR

2 FOR RIGHT CURVE

? 0

WHAT IS CORRESPONDING ELEVATION LEFT SIDE?

? 15.0

WHAT IS CORRESPONDING ELEVATION CENTERLINE?

? 15.0

WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?

? 14.0

WHAT IS THE NEXT STATION NO.?

? 128.0

IS THIS STATION A CURVE PC?

IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR

2 FOR RIGHT CURVE

? 0

WHAT IS CORRESPONDING ELEVATION LEFT SIDE?

? 15.0

WHAT IS CORRESPONDING ELEVATION CENTERLINE?

? 15.0

WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?

? 14.0

WHAT IS THE NEXT STATION NO.?

? 129.0

IS THIS STATION A CURVE PC?

IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR

2 FOR RIGHT CURVE

? 0

WHAT IS CORRESPONDING ELEVATION LEFT SIDE?

? 16.0

WHAT IS CORRESPONDING ELEVATION CENTERLINE?

? 16.0

WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?

? 16.0

WHAT IS THE NEXT STATION NO.?

? 129.72

IS THIS STATION A CURVE PC?

IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR

2 FOR RIGHT CURVE

? 2

WHAT IS CORRESPONDING ELEVATION LEFT SIDE?

? 16.0

WHAT IS CORRESPONDING ELEVATION CENTERLINE?

? 16.7

WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?

? 18.0

WHAT IS THE NEXT STATION NO.?
 ? 130.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 17.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 17.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 15.0

 WHAT IS THE NEXT STATION NO.?
 ? 131.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 17.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 17.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 16.0

 WHAT IS THE NEXT STATION NO.?
 ? 132.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 17.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 19.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 17.0

 WHAT IS THE NEXT STATION NO.?
 ? 133.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 10.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 20.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 17.0

WHAT IS THE NEXT STATION NO.?
 ? 134.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 19.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 21.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 19.0

 WHAT IS THE NEXT STATION NO.?
 ? 135.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 20.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 20.5
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 21.0

 WHAT IS THE NEXT STATION NO.?
 ? 136.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 20.5
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 20.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 21.0

 WHAT IS THE NEXT STATION NO.?
 ? 137.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 19.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 19.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 20.0

WHAT IS THE NEXT STATION NO.?
 ? 138.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 18.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 18.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 17.0

 WHAT IS THE NEXT STATION NO.?
 ? 139.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 17.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 18.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 18.0

 WHAT IS THE NEXT STATION NO.?
 ? 140.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 17.0
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 18.0
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 18.0

 WHAT IS THE NEXT STATION NO.?
 ? 141.0
 IS THIS STATION A CURVE PC?
 IF NO ENTER 0, IF YES ENTER 1 FOR LEFT CURVE OR
 2 FOR RIGHT CURVE
 ? 0
 WHAT IS CORRESPONDING ELEVATION LEFT SIDE?
 ? 18.5
 WHAT IS CORRESPONDING ELEVATION CENTERLINE?
 ? 18.5
 WHAT IS CORRESPONDING ELEVATION RIGHT SIDE?
 ? 17.5

STA NO	LS ELEV	CL ELEV	RS ELEV	SLOPE
127.000	15.000	15.000	14.000	0.000
128.000	15.000	15.000	14.000	0.000
129.000	16.000	16.000	16.000	1.000
129.920	16.000	16.900	18.000	.978
130.000	17.000	17.000	15.000	1.250
131.000	17.000	17.000	16.000	0.000
132.000	17.000	19.000	17.000	2.000
133.000	10.000	20.000	17.000	1.000
134.000	19.000	21.000	19.000	1.000
135.000	20.000	20.500	21.000	-.500
136.000	20.500	20.000	21.000	-.500
137.000	19.000	19.000	20.000	-1.000
138.000	18.000	18.000	19.000	-1.000
139.000	17.000	18.000	18.000	0.000
140.000	17.000	18.000	18.000	0.000
141.000	18.500	18.500	17.500	.500

YOU MUST REVIEW THE SLOPE AND DETERMINE THE DESIRED GRADE OR GRADES.

WHAT IS THE GRADE FROM 127.00 TO 128.00

? 1.0

WHAT IS THE GRADE FROM 128.00 TO 129.00

? 1.0

WHAT IS THE GRADE FROM 129.00 TO 129.92

? 1.0

WHAT IS THE GRADE FROM 129.92 TO 130.00

? 1.0

WHAT IS THE GRADE FROM 130.00 TO 131.00

? 1.0

WHAT IS THE GRADE FROM 131.00 TO 132.00

? -0.5

WHAT IS THE GRADE FROM 132.00 TO 133.00

? -0.5

WHAT IS THE GRADE FROM 133.00 TO 134.00

? -0.5

WHAT IS THE GRADE FROM 134.00 TO 135.00
? -0.5
WHAT IS THE GRADE FROM 135.00 TO 136.00
? -0.5
WHAT IS THE GRADE FROM 136.00 TO 137.00
? -0.5
WHAT IS THE GRADE FROM 137.00 TO 138.00
? 0.0
WHAT IS THE GRADE FROM 138.00 TO 139.00
? 0.0
WHAT IS THE GRADE FROM 139.00 TO 140.00
? 0.0
WHAT IS THE GRADE FROM 140.00 TO 141.00
? 0.0

WHAT IS THE FINAL GRADE CENTERLINE ELEVATION
UP THE FIRST STATION ?
? 15.0

A VERTICAL CURVE IS NECESSARY AT STA 131.00
HOW MANY STATIONS DO YOU WANT TO INCLUDE IN THE CURVE ?
? 2.0

A VERTICAL CURVE IS NECESSARY AT STA 137.00
HOW MANY STATIONS DO YOU WANT TO INCLUDE IN THE CURVE ?
? 4.0

***** CROSS SECTION *****

STA	BACKL	UTIL	SHL	FVHL	CL
127.000	16.209	13.982	14.265	14.517	15.000
128.000	17.209	14.982	15.265	15.517	16.000
129.000	18.209	15.982	16.265	16.517	17.000
129.920	19.129	16.902	17.185	17.457	17.920
130.000	19.209	16.982	17.265	17.517	18.000
131.000	19.834	17.607	17.890	18.142	18.625
132.000	19.709	17.482	17.765	18.017	18.500
133.000	19.209	16.982	17.265	17.517	18.000
134.000	18.709	16.482	16.765	17.017	17.500
135.000	18.209	15.982	16.265	16.517	17.000
136.000	17.772	15.544	15.828	16.080	16.565
137.000	17.459	15.232	15.515	15.767	16.250
138.000	17.272	15.044	15.328	15.580	16.065
139.000	17.209	14.982	15.265	15.517	16.000
140.000	17.209	14.982	15.265	15.517	16.000
141.000	17.209	14.982	15.265	15.517	16.000

STA	FVHLR	SHR	DIR	BACKR
127.000	14.517	14.265	12.767	14.994
128.000	15.517	15.265	13.767	15.994
129.000	16.517	16.265	14.767	16.994
129.920	17.437	17.185	15.687	17.914
130.000	17.517	17.265	15.767	17.994
131.000	18.142	17.890	16.392	18.619
132.000	18.017	17.765	16.267	18.494
133.000	17.517	17.265	15.767	17.994
134.000	17.017	16.765	15.267	17.494
135.000	16.517	16.265	14.767	16.994
136.000	16.080	15.828	14.329	16.557
137.000	15.767	15.515	14.017	16.244
138.000	15.580	15.328	13.829	16.057
139.000	15.517	15.265	13.767	15.994
140.000	15.517	15.265	13.767	15.994
141.000	15.517	15.265	13.767	15.994

.....
SUBROUTINE INSECT COMPUTES THE INTERSECTION OF THE
SIDESLOPES WITH THE EXISTING GROUND AT EACH STA.
INPUT A SLOPE GRADE IN RESPONSE TO THE PROGRAM'S QUESTIONS.
USE ANY WHOLE NUMBER UP TO TWO DIGITS.
I.E. USE 2 FOR 1:2 OR 4 FOR 1:4

WHAT IS THE RIGHT SLOPE GRADE AT STA. 127.00?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 128.00?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 129.00?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 129.92?
? 3

WHAT IS THE RIGHT SLOPE GRADE AT STA. 130.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 131.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 132.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 133.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 134.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 135.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 136.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 137.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 138.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 139.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 140.00?
? 2

WHAT IS THE RIGHT SLOPE GRADE AT STA. 141.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 127.00?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 128.00?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 129.00?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 129.92?
? 3

WHAT IS THE LEFT SLOPE GRADE AT STA. 130.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 131.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 132.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 133.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 134.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 135.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 136.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 137.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 138.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 139.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 140.00?
? 2

WHAT IS THE LEFT SLOPE GRADE AT STA. 141.00?
? 2


```

===== INTERSECTION POINTS =====

```

STA NO	LEFT HAND SIDESLOPE		RIGHT HAND SIDESLOPE	
	GRADE	ELEV	GRADE	ELEV
127.00	3	15.00	3	13.93
128.00	3	15.00	3	13.89
129.00	3	16.00	3	16.00
129.92	3	15.87	3	18.91
130.00	2	17.00	2	14.72
131.00	2	17.00	2	15.89
132.00	2	16.79	2	16.86
133.00	2	4.66	2	16.79
134.00	2	19.71	2	19.49
135.00	2	20.21	2	21.99
136.00	2	20.77	2	21.56
137.00	2	19.46	2	20.24
138.00	2	18.27	2	20.06
139.00	2	18.21	2	18.99
140.00	2	18.21	2	18.99
141.00	2	19.21	2	17.99

```

=====
SUBROUTINE XSECT COMPUTES CROSS SECTIONAL AREAS
FOR EACH STA, VOLUME BETWEEN STAS, AND CUMULATIVE
VOLUME. FILL AREAS ARE GIVEN A + SIGN AND CUT AREAS
ARE INDICATED BY A - SIGN.
=====

```


STA NO	CROSS SECTION AREAS(SF)	VOLUME BETWEEN STATIONS(CY)	CUMULATIVE VOLUME(CY)
127.00	-10.69	0.00	0.00
128.00	56.81	85.41	85.41
129.00	38.28	176.10	261.52
129.92	29.02	114.66	376.18
130.00	69.55	14.60	390.78
131.00	90.88	297.11	687.89
132.00	-4.87	159.29	847.18
133.00	-103.02	-199.79	647.39
134.00	-196.05	-553.65	93.54
135.00	-265.06	-853.92	-760.38
136.00	-279.15	-1007.80	-1768.18
137.00	-221.93	-927.93	-2696.11
138.00	-179.45	-743.30	-3439.41
139.00	-165.46	-638.72	-4078.13
140.00	-165.46	-612.82	-4690.94
141.00	-180.30	-640.29	-5331.23

SUBROUTINE MASHAL FINDS MAXIMUM AND MINIMUM CUMULATIVE VOLUMES, AND IDENTIFIES THEIR STATION LOCATION POINTS. IT ALSO PLOTS A MASS DIAGRAM ON THE VERSATEC PLOTTER, NO INPUT IS NECESSARY.

THE MAXIMUM CUMULATIVE VOLUME IS 847.18 CUBIC YARDS THIS OCCURS AT STATION 127.00

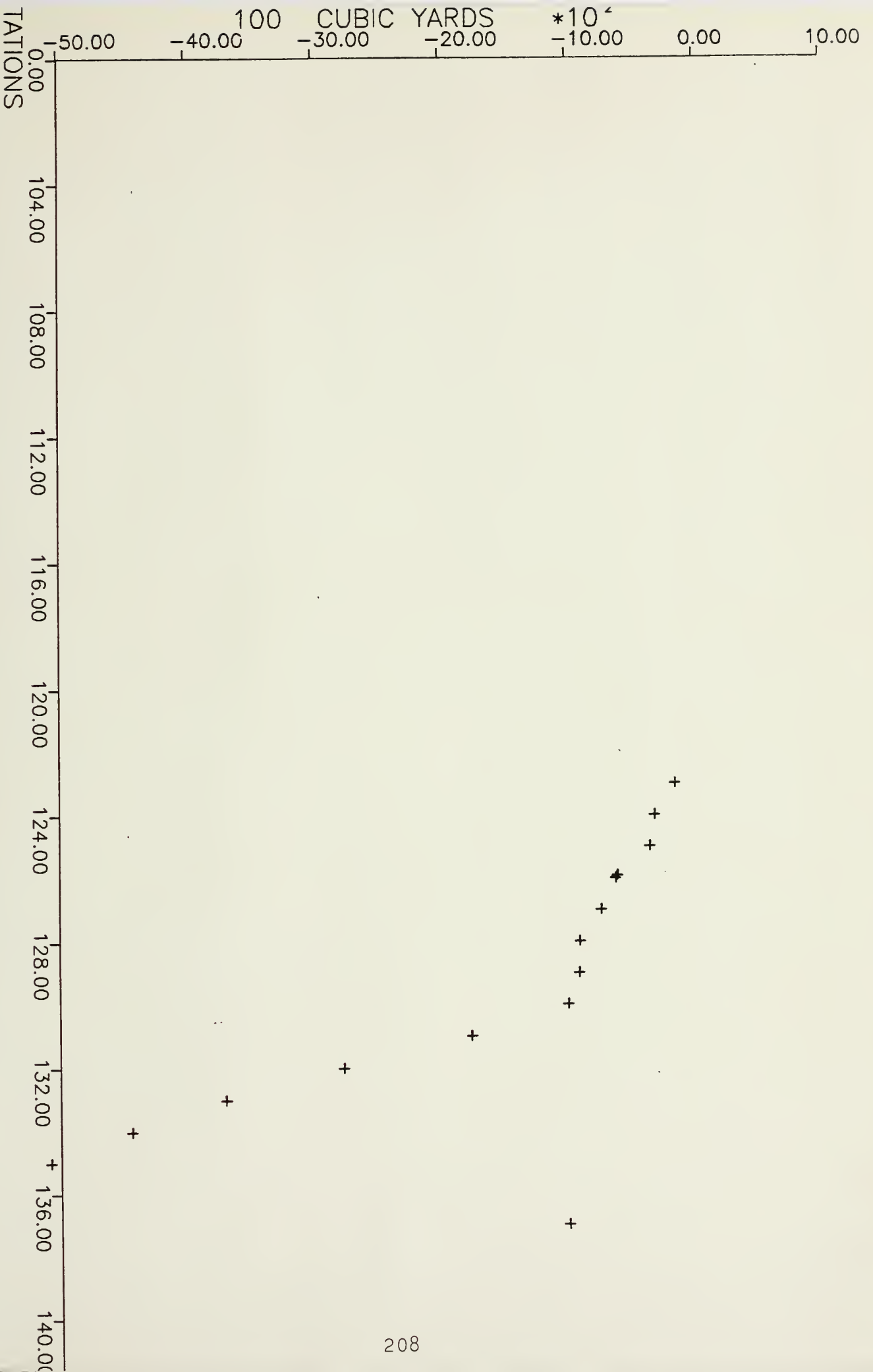
THE MINIMUM CUMULATIVE VOLUME IS --5551.25 CUBIC YARDS
THIS OCCURS AT STATION 141.00

GO TO THE PLOTTER BIN AND REMOVE OUTPUT (MSS DIAGRAM)

=====

THIS PROGRAM WILL END
5.252 CP SECONDS EXECUTION TIME.
/PLOT(CV(TAPEY)
PLOT. 81/11/03, 12.21.55.
/BYE

CE513AA L06 OFF 12.23.00.
CE513AA SRU 54.444 UNITS.



Thesis
C9255
c.1

Thesis
C9255
c.1

Cuccu

FORTRAN interactive
road design.

202531

202531

Thesis
C9255
c.1

Cuccu

FORTRAN interactive
road design.

thesC9255

FORTRAN interactive road design.



3 2768 002 09821 2

DUDLEY KNOX LIBRARY